DTIC FILE COPY

RADC-TR-90-50
Final Technical Report
April 1990

AD-A227 706

OPTICAL MATRIX INVERTER FOR PHASED ARRAY RADAR

University of Alabama in Huntsville

Mustafa A.G. Abushagur

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

Rome Air Development Center
Air Force Systems Command
Griffiss Air Force Base, NY 13441-5700

This report has been reviewed by the RADC Public Affairs Division (PA) and is releasable to the National Technical Information Services (NTIS) At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-90-50 has been reviewed and is approved for publication.

APPROVED:

DAVID A. CORDEIRO
Project Engineer

APPROVED:

JOHN A. GRANIERO
Technical Director
Directorate of Communications

FOR THE COMMANDER:

JAMES W. HYDE III
Directorate of Plans & Programs

DEPARTMENT OF THE AIR FORCE
HEADQUARTERS ROME AIR DEVELOPMENT CENTER (AFSC)
GRIFFISS AIR FORCE BASE, NEW YORK 13441-5700

REPLY TO ATTN OF: XP (B.G. Oaks, DSN: 587-7052)

25 SEP 1990

SUBJECT: Technology Screening of Unclassified/Unlimited Reports

TO: Defense Logistics Agency
Defense Technical Information Center
Cameron Station
Alexandria, VA 22304-6145
Attn: DTIC-HAS

1. Reference: DTIC Letter, subject as above. Undated referenced letter questioned the public releasability of the following report:

Source: University of Alabama in Huntsville
Title: Optical Matrix Inverter for Phased Array Radar
Report No. RADC-TR-90-50
Date of Report: April 1990

2. RADC has conducted an extensive review of the report against the MCTL as was initially done. Our review verified the original release recommendation as appropriate for the following reasons:

a. During the contract the device was strictly laboratory experimental.

b. The advances in technology resulting from the work were very small (no technological breakthroughs).

c. While the work was supported with 6.3A funds, it was essentially fundamental research as defined in the former Secretary of Defense policy memorandum which stated that fundamental research conducted wholly on a university campus by university researchers would be public releasable.

d. The architecture of the design worked on in this contract was not new nor did it display any novel ideas.

3. It is important to note that the appendices of the report contain papers on the subject that were presented in public forums by the researchers.

4. For the above reasons we continue to recommend unlimited public release for the report. Should you have any further questions, please contact the RADC MCTL Focal Point, Mr Bill Oaks, DSN 587-7052.

John A. Ritz

John A. Ritz, Acting Director
Directorate of Plans & Programs

REPORT DOCUMENTATION PAGE

Form Approved OPM No. 0704-0188

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	April 1990	Final Apr 88 - Sep 88

4. TITLE AND SUBTITLE

OPTICAL MATRIX INVERTER FOR PHASED ARRAY RADAR

5. FUNDING NUMBERS

C - F30602-81-C-0185
PE - 63726F
PR - 2863
TA - 92
WU - PB

6. AUTHOR(S)

Mustafa A. G. Abushagur

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Alabama in Huntsville
Electrical & Computer Engineering Dept.
Huntsville AL 35899

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Rome Air Development Center (DCLW)
Griffiss AFB NY 13441-5700

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

RADC-TR-90-50

11. SUPPLEMENTARY NOTES

RADC Project Engineer: David A. Cordeiro/DCLW/(315) 330-4092

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The purpose of this study is to investigate new approaches to process the data collected by phased array antennas to achieve interference cancelling. This data processing needs to be performed with severe time constraints. A hybrid optoelectronic processor, the bimodal optical computer (BOC), was introduced. The BOC was implemented to process the phased array data. The BOC algorithm was analyzed for the convergence of the optimal weights. Convergence dependence on the errors involved in the analog optics devices, the condition number of the matrix and the strength of the jammer signal were considered. An experimental model of the BOC was built and used in solving systems of linear equations and matrix inversion. An agreement between theory and experiments is shown. Computer simulation of the BOC in interference cancelling was presented. Phased Arrays; Radar

14. SUBJECT TERMS

Phased Arrays, Optical Algebraic Processors, Optical Matrix Inverter, Optical Computing

15. NUMBER OF PAGES	16. PRICE CODE
64	

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

Optical Matrix Inverter for Phased Array Radar

I. Introduction

The purpose of this study is to investigate new approaches to process the data collected by phased array antennas to achieve interference canceling. This data processing need to be performed with severe time constraints. This study was motivated by the fact that optics is capable of processing data in parallel with very high speeds. The issue we investigated in this study was the applicability of the optical algebraic processors to such problems. The advantages gained by using optical systems over their electronic counterparts are investigated. The limitations of the optical system and their effect on the interference canceling was studied. In the course of this contract we investigated the issues just listed and came to valid and encouraging conclusions which are outlined in the final section of this report.

II. Interference Canceling

Phased array antennas are used both in radar and communication systems among many other applications. These systems are used to detect a desired signal. In practical systems the desired signal will not be the only signal present. Thermal noise, other friendly and unfriendly interference signals usually are present in surrounding environment. It is common that the jamming signals be stronger than the desired signal, which presents a serious problem in detecting the desired signal. It was suggested, to resolve this problem, that if we can change the antenna pattern in such a way that we introduce a null along the direction of the jammer, it will cancel it without effecting the ability of detecting the desired signal [1–5]. This change of the antenna pattern can be achieved by adapting the weights of the individual sensors to the environment. This adaptation process needs to be done using the detected signals. This is the basic idea behind the adaptive phased array antenna systems. In the following section we show how this adaptation processes can be achieved.

2.1 Adaptive Phased Array Antennas

In adaptive phased arrays the incoming signal is detected by an array of sensors. The detected signal is a combination of the target signal plus interference and noise signals. The system is adjusted in such a way to suppress the interference signals reception without affecting the desired signal.

In this section we consider the two general cases of interference canceling: first by assuming that the interference signal direction is known; secondly by assuming no *a priori* information is known about the interference signal.

A. Interference Signal Direction is Known

When the interference signal direction is known the weights w_i's of the array can be chosen to suppress the interference signal. Let the system shown in Fig. 1(a) be used to demonstrate this adaptation technique. The output signal of the array s(t) is given by [1]

$$\begin{aligned} s(t) = {} & P[(w_1 + w_3)\sin\omega_0 t + (w_2+w_4)\sin(\omega_0 t - \theta - \tfrac{\pi}{2})] \\ & + I[w_1 \sin(\omega_0 t - \theta) + w_2 \sin(\omega_0 t - \theta - \tfrac{\pi}{2}) \\ & + w_3 \sin(\omega_0 t + \theta) + w_4 \sin(\omega_0 t + \theta - \tfrac{\pi}{2})]\,, \end{aligned} \tag{1}$$

where

P = the pilot signal,

I = the interference signal, and

θ = the phase shift

$$\theta = \frac{2\pi d}{\lambda}\sin\psi. \tag{2}$$

To cancel the interference signal and to make the signal s(t) equal to the pilot signal, we need to solve the following system of linear equations for the weights w_i's:

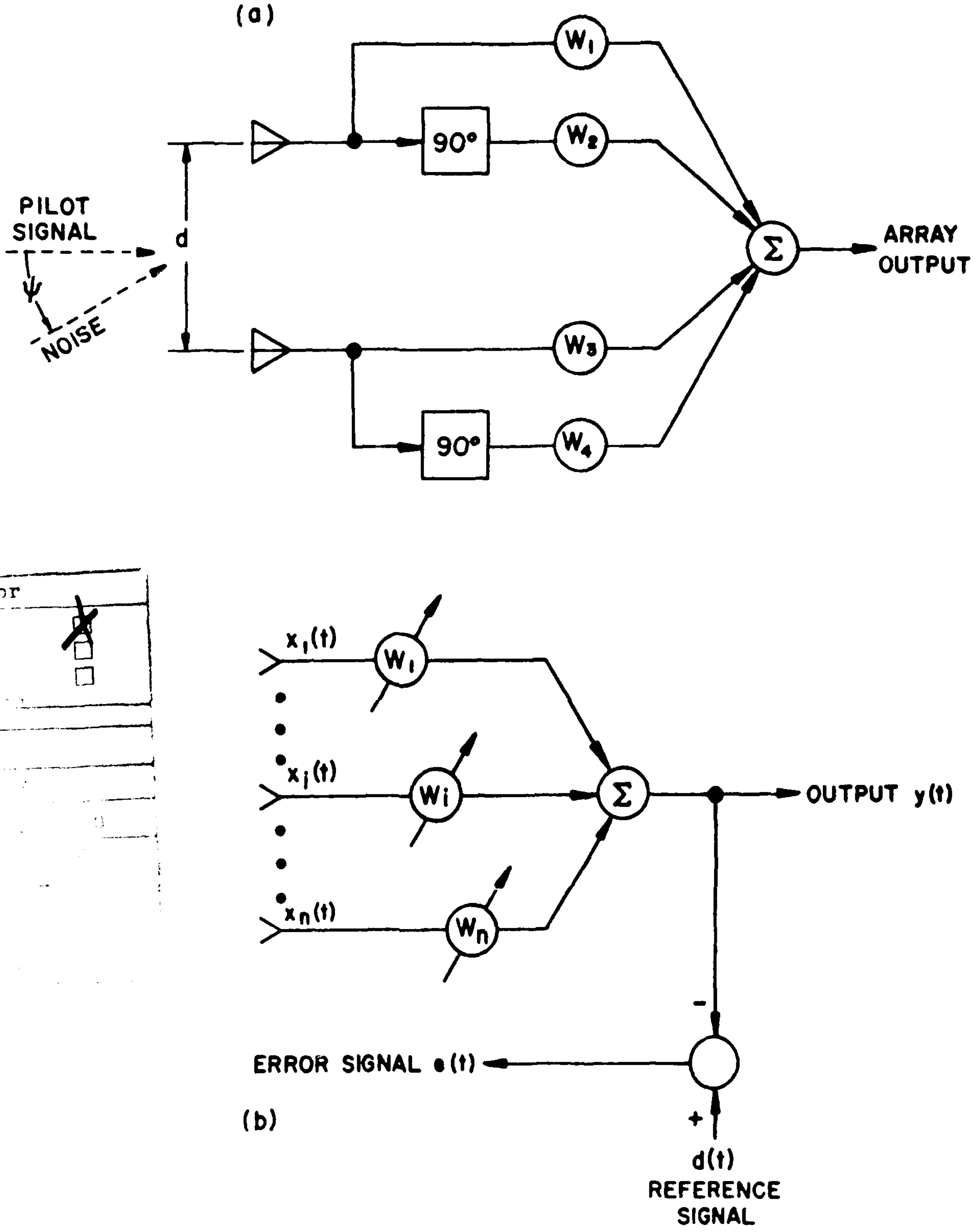


Figure 1 Basic adaptive array system with (a) signal and noise directions are known, and (b) no *a priori* information is assumed.

$$\left.\begin{aligned} w_1 + w_3 &= 1 \\ w_2 + w_4 &= 0 \\ (w_1 + w_3)\cos\theta - (w_2 - w_4)\sin\theta &= 0 \\ (w_2 + w_4)\cos\theta + (w_1 - w_3)\sin\theta &= 0 \;. \end{aligned}\right\} \tag{3}$$

The size of this system of linear equations depends on the number of sensors in the array. The number of jammers can make the system under or overdetermined, which are both time consuming algebra problems.

B. No *a priori* Information is Known

This is the most general case where we assume no information about jammers. The system used in this case is shown in Fig. 1(b). Each of the n sensors receives a signal $x_i(t)$ which is in turn multiplied by a variable weight w_i. The output signal s(t) is compared with the desired signal d(t) and their difference, the error signal $\epsilon(t)$, is used to determine the value of w_i's. The output of the array is

$$s(t) = \sum_{i=1}^{n} x_i(t)\, w_i \tag{4}$$

or

$$s(t) = \vec{w}^T \vec{x} \tag{5}$$

where

$$\vec{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_i \\ \vdots \\ w_n \end{bmatrix} \text{ and } \vec{x} = \begin{bmatrix} x_1(t) \\ \vdots \\ x_i(t) \\ \vdots \\ x_n(t) \end{bmatrix} . \tag{6}$$

For digital sampled data

$$s(j) = \vec{w}^T \vec{x}(j) \;, \tag{7}$$

and

$$\epsilon(j) = d(j) - s(j) = d(j) - \vec{w}^T \vec{x}(j) \;. \tag{8}$$

The optimum value of the weights, w_i's, is the one reduces $\epsilon(j)$ to zero or at least minimize it.

For N samples of data the optimum weights satisfy the following set of systems of linear equations:

$$\left.\begin{array}{l} \vec{w}^T\,\vec{x}(1) = d(1) \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ \vec{w}^T\,\vec{x}(i) = d(i) \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ w^T\,\vec{x}(N) = d(N)\;. \end{array}\right\} \tag{9}$$

The N sets of equations have n unknowns, and usually N >> n, and are inconsistent and over specified. The optimization problem can be rewritten as

$$\vec{w}_{opt} = R_{xx}^{-1}\,\vec{r}_{xd} \tag{10}$$

where

$$R_{xx} = E\{\vec{x}\vec{x}^T\}\;, \tag{11}$$

and

$$\vec{r}_{xd} = E\{\vec{x}d\}. \tag{12}$$

The matrix R_{xx} is called the covariance matrix, where $E\{\cdot\}$ is the ensemble average.

Many algorithms are introduced [2] to solve for the weights in Eq. (10). Some of the popular algorithms are the least mean square (LMS), and the direct matrix inversion (DMI).

We'll briefly mention the DMI algorithm since it leads to the algorithm introduced in this paper. Eq. (10) cannot be determined exactly using a limited number of samples of the input data. For practical consideration a small number of samples is detected to be used in determining $\vec{w}$. The estimated value of Eq.(10) can be given by

$$\hat{\vec{w}} = \hat{R}_{xx}^{-1}\,\hat{\vec{r}}_{xd}\;. \tag{13}$$

where

$\hat{R}_{xx}$ is the sample covariance matrix, and $\hat{\vec{r}}_{xd}$ is the sample cross–correlation vector, and are given by

$$\hat{R}_{xx} = \frac{1}{K} \sum_{j=1}^{K} \vec{x}(j)\, \vec{x}^{T}(j) \tag{14}$$

and

$$\hat{\vec{r}}_{xd} = \frac{1}{K} \sum_{j=1}^{K} \vec{x}(j)\, d(j) \ , \tag{15}$$

and K is the number of samples. The DMI algorithm determines the inverse of the sample covariance matrix $\vec{R}_{xx}$, then from Eq. (13) evaluates $\hat{\vec{w}}$.

III. Optical Algebraic Processors

Solving systems of linear equations and determining the eigenvalue and the eigenvectors are only a few of the challenging problems faced by the numerical computations. The problem of determining the weights for the adaptive phased array as given by Eq. (13) is solving a system of linear equations. In this case the size of the matrices involved are very large. Solving a system of linear equations for large matrices is time consuming because of the computation complexity. Digital computers revolutionized this field, because of the fast execution of number crunching operations. But still solving a problem with a matrix of 1000×1000 elements takes few seconds, which by our standards, is a long time.

Optics by its inherent parallelism and speed seems to present a natural choice for solving this class of problems. Analog optics is very attractive for optical information processing and computing. As shown in Fig. 2, in the vector–matrix multiplier all the elements of the vector are processed in the same time. At almost the same time we write $\vec{x}$ we do read $\vec{b}$. If the optical path length between the input and output planes is 3cm, the whole operation of the vector–matrix operation can be done in less than 100psec. For N=1000, the number of operations needed to perform $A\vec{x}-\vec{b}$ is $0(10^{6})$. Hence, speed of the processor is $0(10^{-16})$ sec/operation. This illustrative example gives a sense of the speed of

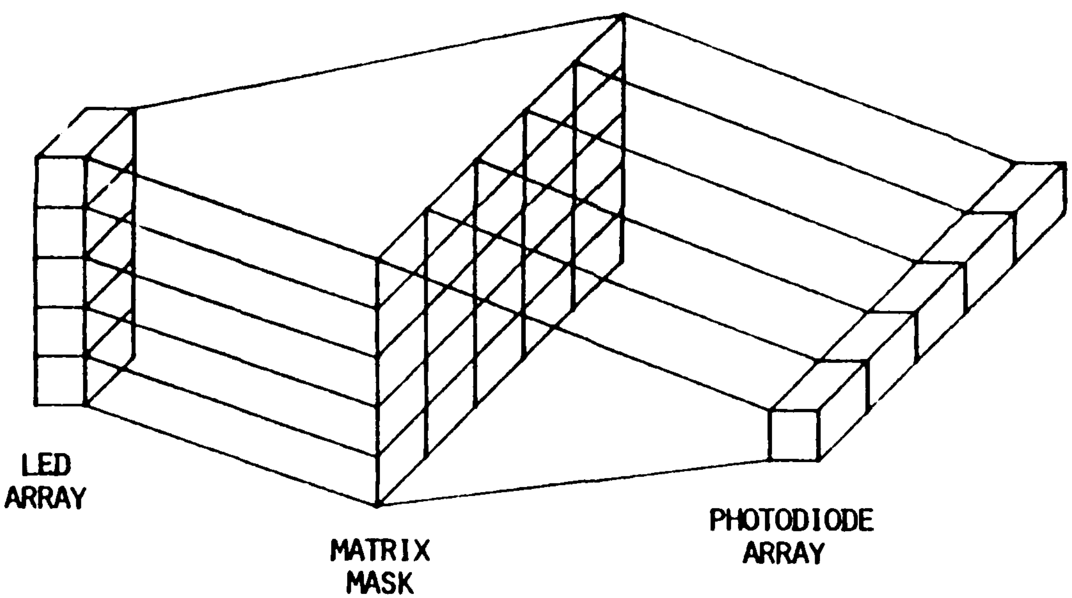


Figure 2 **Incoherent vector-matrix multiplier**

the analog optics in performing linear algebra operations. Unfortunately this high speed of operations is combined with a low accuracy, which is the nature of the all analog systems.

Analog optics is very fast but inaccurate. On the other hand, digital electronics is very accurate but not as fast as analog optics. Utilizing the advantages of both analog optics and digital electronics can be achieved in a "compromise" hybrid system. A system that slows down the processor speed but in return increases the accuracy substantially.

The bimodal optical computer (BOC) introduced by Caulfield, et. al. [6] is based on this idea of combining the speed of analog optics and the accuracy of digital electronics.

The adaptation problem for the weights $\vec{w}$ introduced in Section II, can be rewritten in the following form, from Eq. (13)

$$\hat{R}_{xx}\,\hat{\vec{w}} = \hat{\vec{r}}_{xd} \ , \tag{16}$$

which can be written as

$$A\vec{x} = \vec{b} \ , \tag{17}$$

where $A = \hat{R}_{xx}$,

$\vec{x} = \hat{\vec{w}}$,

and $\vec{b} = \hat{\vec{r}}_{xd}$.

Eq. (16) is a system of linear equations can be solved using the bimodal optical computer.

Consider an N×N matrix A, and N×1 vectors x and b. Let A and $\vec{b}$ are given, and we would like to solve the system of equations given by Eq. (17) for the vector $\vec{x}$. This can be solved by analog optics techniques. The relaxation method introduced by Cheng an Caulfield [7] can be used to solve Eq. (17) for $\vec{x}$. Consider the hybrid system shown in Fig. 3. Assume an initial value for the solution $\vec{x}$ and write it using the LED's. Then the vector $\vec{x}$ is multiplied by the matrix A. The resultant vector $\vec{y}$ is compared with $\vec{b}$ by a difference amplifier. This difference is fed back to correct $\vec{x}$. This process of multiplying the new value of $\vec{x}$ with A and comparing $\vec{y}$ to $\vec{b}$ continues till the difference between $\vec{y}$ and

$\vec{b}$ becomes zero. Then the value of $\vec{x}$ will converge to the solution of Eq. (17). For a positive definite matrix A always a convergence to the solution exists. To achieve a nonnegative definite matrix, we can multiply Eq. (17) from the left by the Hermitian A^H of A. The new $A^H A$ is non–nonnegative definite. We will show later that the increase in condition number this causes need not affect convergence and that we can solve the equations even if $A^H A$ is singular.

This method in solving a system of linear equations is very rapid. Its speed is limited only by the speed of the electro–optics devices and on the feedback electronics, which can be in the psec range.

Let us consider now the accuracy of the system. In writing both A and $\vec{x}$ on the optical mask (it can be a photographic film or a spatial light modulator) and the LED array, a considerable amount of error will exist because of the nature of these analog devices. Also reading the vector $\vec{b}$ on the photodiode array cannot be done exactly. Therefore, the system in Fig. 3 did not solve the system in Eq. (17) but instead the system given by

$$A_0 \vec{x}_0 = \vec{b}_0. \tag{18}$$

where the subscript zeros indicate inaccuracies in the optics and electronics. The solution $\vec{x}_0$ of Eq. (18) can be refined to get the vector $\vec{x}$ using the following algorithm:

(a) Solve the system in Eq.(18) using the analog optical processor for $\vec{x}_0$.

(b) Store the solutions $\vec{x}_0$ to a high accuracy with the digital processor. Use a dedicated digital processor to calculate the residue

$$\vec{r} = \vec{b} - A\vec{x}_0 = A(\vec{x} - \vec{x}_0) = A\ \Delta\vec{x}\ . \tag{19}$$

(c) Use the optical analog processor to solve the new system of linear equations

$$A_0 \vec{y}_0 = s\vec{r}_0, \tag{20}$$

where $\vec{y} = s\ \Delta\vec{x}$ and s is a "radix," or scale factor, chosen to good use of the dynamic range of the system.

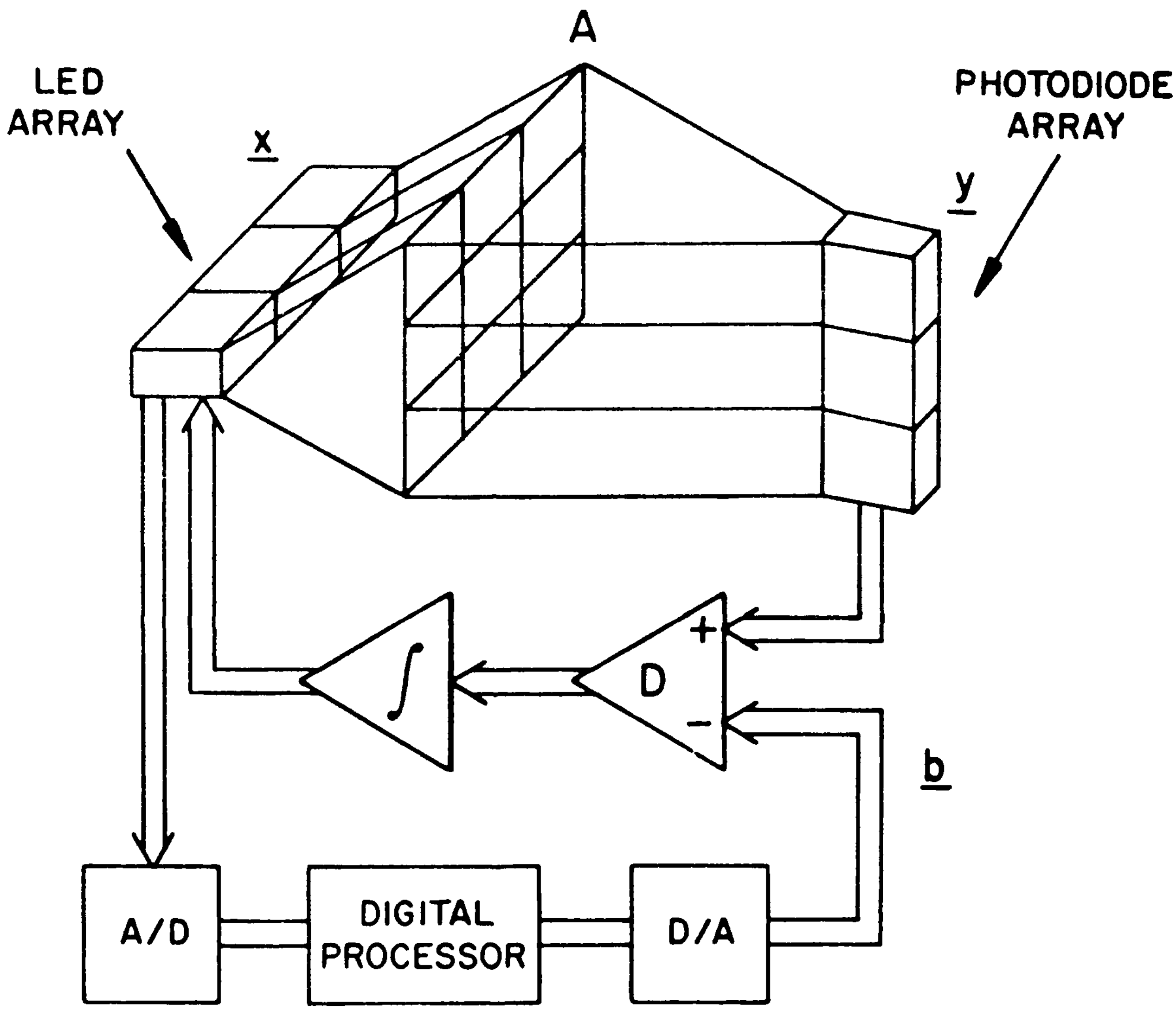


Figure 3 The bimodal optical computer used in solving a system of linear equations.

(d) Use the digital processor to refine the solution $\vec{x}_0$ for $\vec{x}_1$:

$$\vec{x}_1 = \vec{x}_0 + \Delta\vec{x}. \tag{21}$$

If the refined solution $\vec{x}_1$ is accurate enough, terminate the iterations. Otherwise, return to (b), (c), and (d) for a more refined solution. This system which implements the algorithm outlined above is shown in Fig. 3.

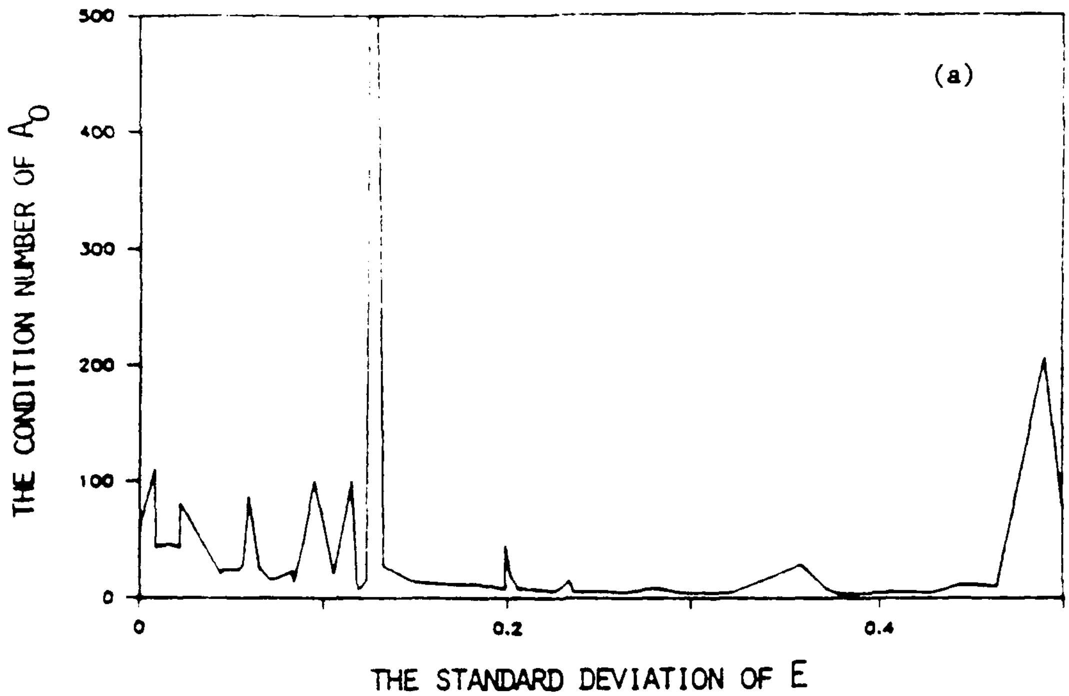
The convergence and speed of the solution for the system of linear equations is studied and reported by Abushagur and Caulfield [8]. The convergence of the iterative solution depends on two main factors. First, is the condition number of the matrix $A_0, \chi(A_0)$. The smaller condition number the faster it will converge. Secondly, on the error involved in reading and writing of A, $\vec{x}$ and $\vec{b}$ using the electro–optic devices. The higher the accuracy in representing these parameters the faster the convergence will occur.

The condition number of the matrix is a critical factor in the convergence of the solution. In representing the matrix A by an optical mask, an error will be added to it. The inaccuracies in representing the matrix A changes the values of the matrix elements a_{ij}'s. These variations in the matrix elements change the condition number of the matrix. Let us represent the mask's matrix A_0 in the following form

$$A_0 = A + E, \tag{22}$$

where E is an error matrix. The error matrix E is generated using Gaussian statistics, with standard deviation, σ_E.

The effect of the error matrix E on the condition number of the optical mask's matrix is demonstrated in Fig. 4. In Fig. 4(a), a matrix A with condition number of 60 is considered. The coefficients of the matrix are normalized such that the maximum a_{ij} is equal to the unity. The condition number of A_0 plotted as a function of the standard deviation of the error matrix, σ_E. The condition number of the matrix A_0 tends to decrease by the increase of σ_E, especially for large σ_E. In Fig. 4(b) a matrix A with condition number 300 (an ill–conditioned), the condition number decreased significantly



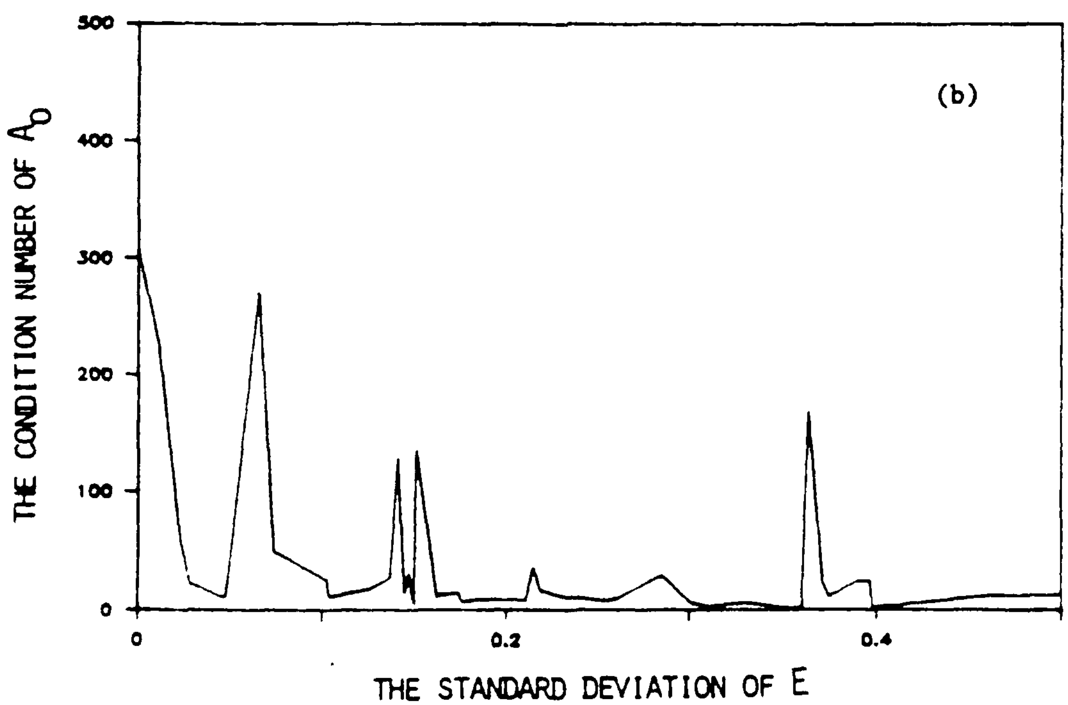


Fig. 4 The condition number of the matrix's mask $\chi(A_0)$, as a function of σ_E, for (a) $\chi(A)=60$, and (b) $\chi(A)=300$.

throughout the range considered of σ_e. Thus, if A is an ill-conditioned matrix, the mask's matrix can very well be a better conditioned one. Of course, in this case when we solve the system give in Eq. (18) A_0 will be different from the original A. Hence, we solve a better conditioned system for the approximate solution $\vec{x}_0$, and then we refine using the algorithm outlined above.

Now, let us consider the effect of the condition number on the convergence of the solution. The condition number is the determining factor in the accuracy of the solution of the system of equations.

Hence, for a matrix with a large condition number, the first iteration of the solution with a limited accuracy computer will be highly inaccurate. This leads to the result that the larger the condition number the larger number of iterations needed for the convergence of the solution. To demonstrate this result, we ran a computer simulation of our bimodal optical computer. The simulated BOC is used to solve a system of linear equation with a 16 bit resolution. The matrix A, was generated randomly using Gaussian statistics. An error matrix E, with an error of 1% of that of the maximum coefficient of the matrix A, and then added to A to generate A_0 as in Eq. (22). An error of 1% also used in reading x_0 and in writing b_0. In each case we computed the condition numbers of the matrix and its mask. The number of iterations required for convergence of the solution to the specified accuracy was determined for each case. The iterations were terminated if they exceeded 25 or $\|\vec{r}^{(k+1)}\|/\|\vec{r}^{(k)}\| > 1$, which is the condition for the solution divergence. The number of iterations, N_I, required for convergence of the solution with 16 bit accuracy is plotted as a function of the condition number $\chi(A_0)$ in Fig. 5. In these experiments it is clear that the number of iteration increases with the increase of the condition number.

The condition number, as shown above, is one of the determining factors for the number of iteration required for convergence of the solution of the system of equations. It is also shown in Fig. 4 that the condition number of the optical mask's matrix decreases by the increase of the standard deviation of the error matrix E.

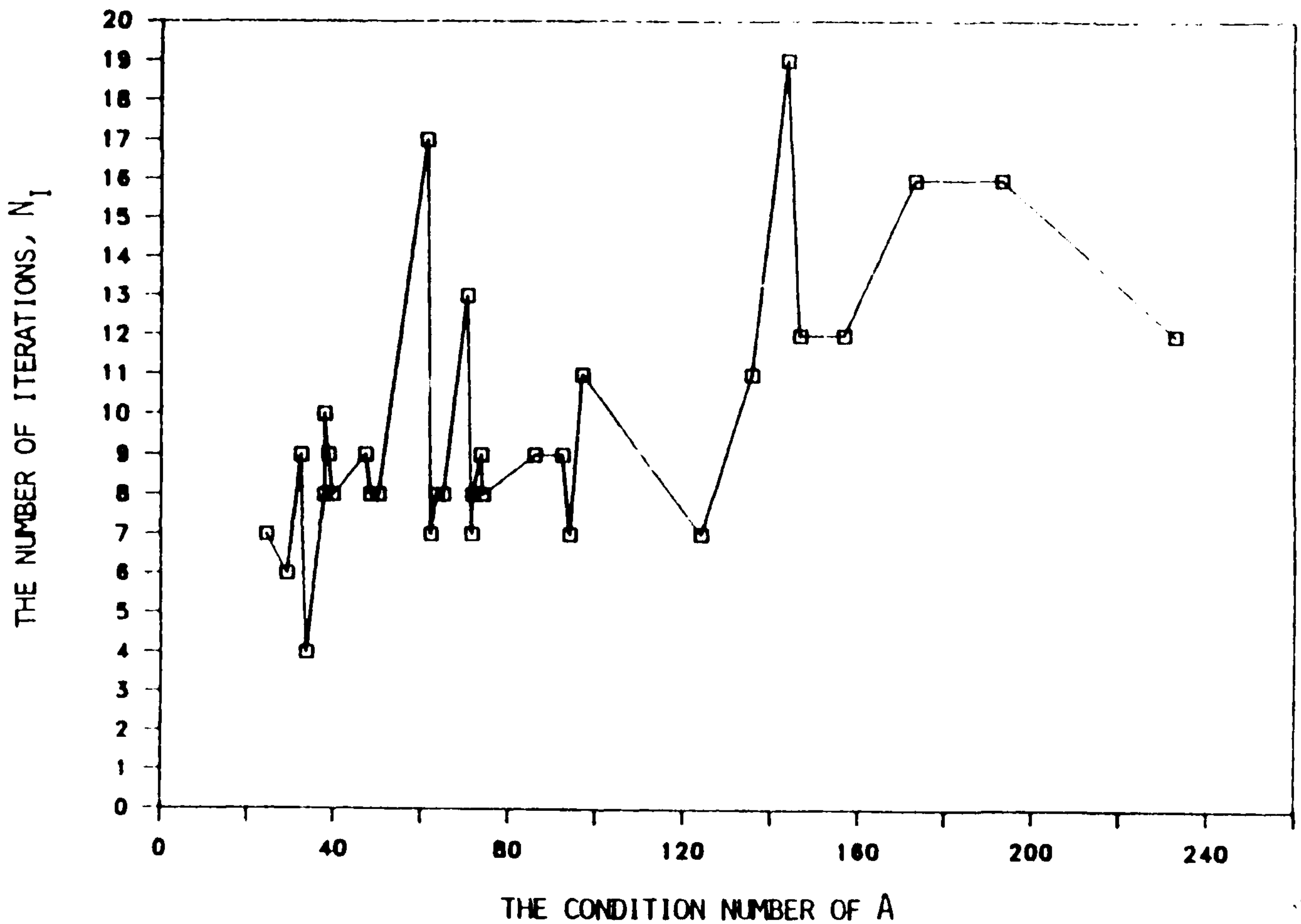


Fig. 5 The number of iterations, N_I, is plotted as a function of the condition number of the matrix, A.

The influence of the standard deviation of the error matrix, E, on the convergence of the solution is shown in Fig. 6. The number of iteration, N_I, increases with the increase of σ_E. This decrease in the convergence rate is expected because for large σ_E's the matrix A_0 is quiet different from A. The important result demonstrated in Fig. 6 is that even with an error up to 50% in writing the matrix A in the optical mask, convergence is still achieved.

The result is very important in realizing this algorithm by analog optics. In representing the matrix A by an optical mask always an error will exist. An error of 1% is quite hard to achieve in this representation using our current technology. In the present state-of-the-art technology an accuracy of 2 to 3% in writing the matrix A is within our reach. This accuracy does not sacrifice the convergence of the solution.

The above results show that the Bimodal Optical Computer can solve a system of linear equations with very high accuracy. This accuracy can be achieved using I/O devices that have limited accuracy. The digital computes are capable of achieving high accuracy solution for all the cases considered above. So, what is the real advantage of introducing this new class of computers? Speed, is what we are after. An analysis of the speed of the BOC shows that for it to be more faster than the digital computer the following condition should be satisfied [8]

$$A_p A_I >> 1, \tag{23}$$

where

$$A_p = \frac{2[N^3/6) + N^2(1-K - NK]}{K} , \tag{24}$$

$$A_I = T_{D1}/T_{A1} , \tag{25}$$

$$K = I_0/I_D , \tag{26}$$

$$N = \text{the size of the matrix} , \tag{27}$$

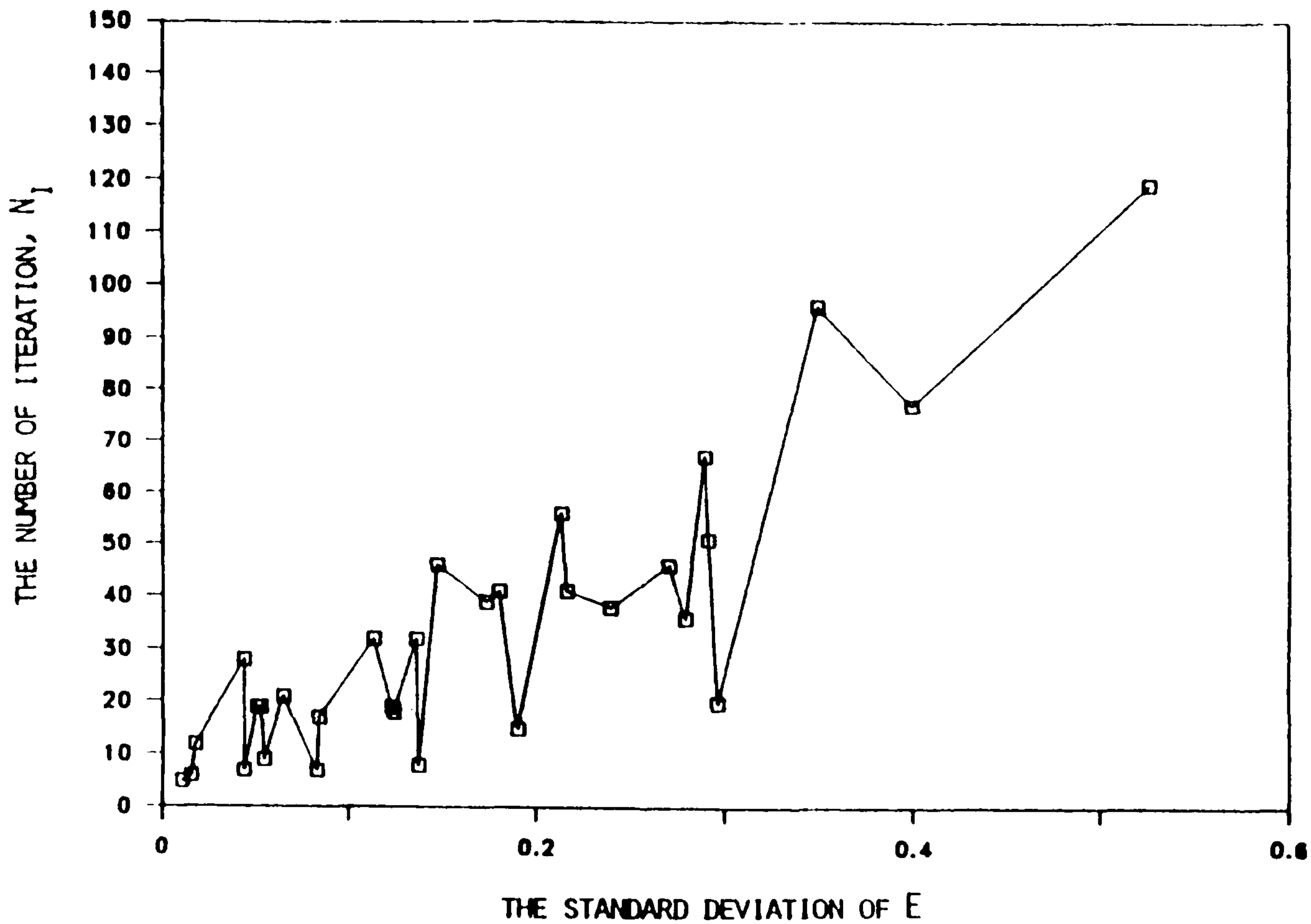


Fig. 6 The number of iterations, N_I, is plotted as a function of the standard deviation, σ_E, of the error matrix.

I_0 = the number of iterations needed for the convergence of the solution to the specified accuracy using the BOC.

I_D = the number of iterations needed for the convergence of the solution using the digital computer.

T_{D1} = the time required to perform one digital operation, and

T_{A1} = the time required to solve $A_0\vec{x}_0 = \vec{b}_0$ using the analog processor.

The speed advantage depends on the size of the matrix, N, and the speed of the electronic and electro–optic devices used in the BOC. The factor A_p is plotted in Fig. 7 as a function of N, for a set of values of K. It is quiet clear that A_p is very large $0(10^5)$ for moderately large values of N.

The values of T_{D1} and T_{A1} can be compared using approximate values using current data.

$$T_{A1} \simeq 2\ \mu\text{sec} \tag{27}$$

$$T_{D1} \simeq 1\ \mu\text{sec for a microcomputer,} \tag{28}$$

and

$$T_{D1} \simeq 1\ \text{nsec for a CRAY2.} \tag{29}$$

The factor $A_p\ A_I$ is plotted in Fig. 8 as a function of N using the data given by Eqs. (27) and (29). The advantage in speed is very large and the condition of eq. (23) is satisfied for N>50.

This advantage in speed of the BOC over the existing digital computer makes it a very attractive computing machine, and shows the potential of this class of hybrid systems.

IV. Impelmenting the BOC

The BOC was built in our laboratory has three main parts as shown in Fig. 3. The optical system, the electronic circuit, and the digital processor. The optical system consists of the fully parallel matrix–vector multiplier. Light from the LED's representing $\vec{x}$ components are spread vertically by planner waveguides onto the columns of the matrix

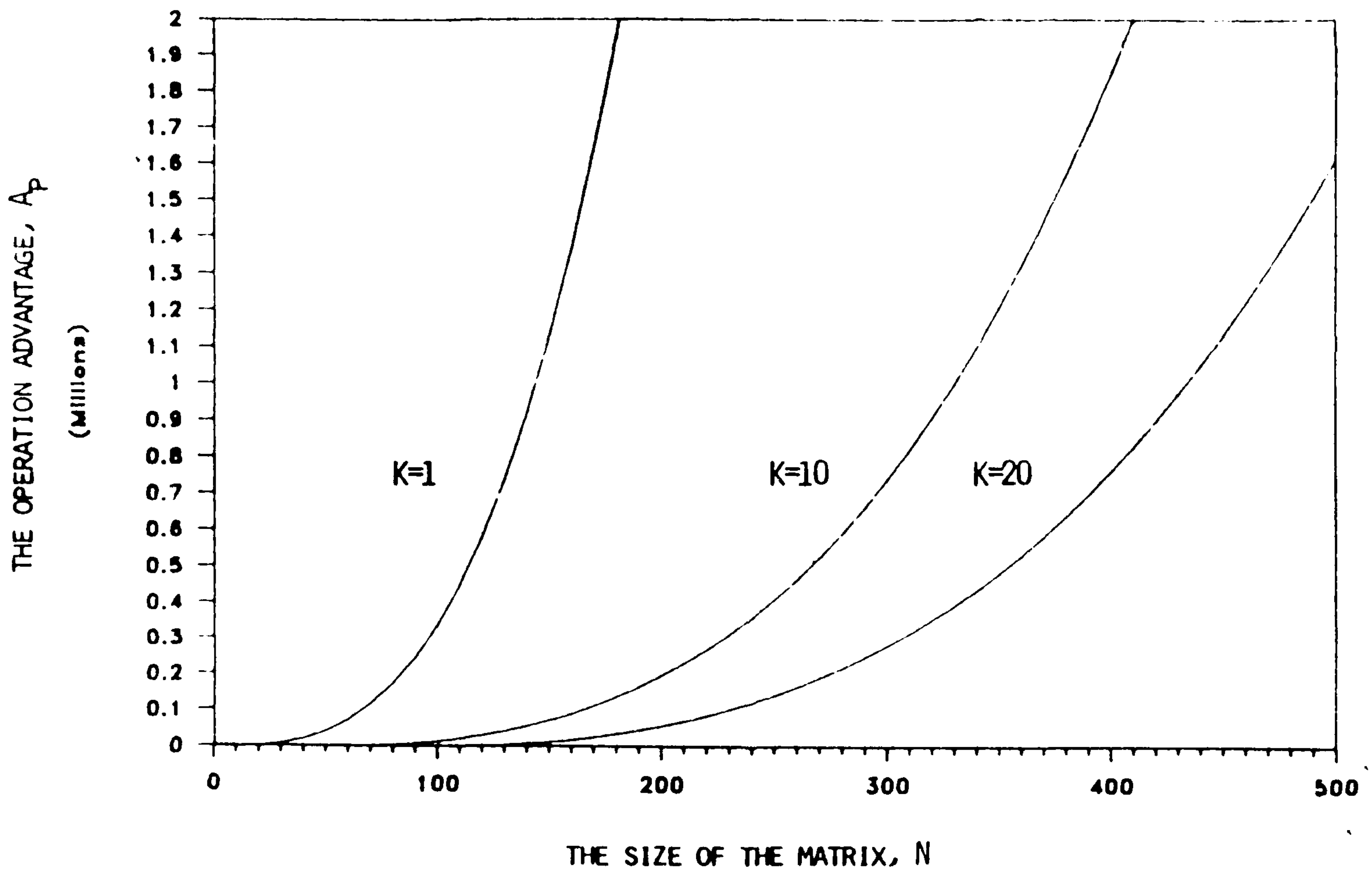


Fig. 7 The operation advantage, A_p.

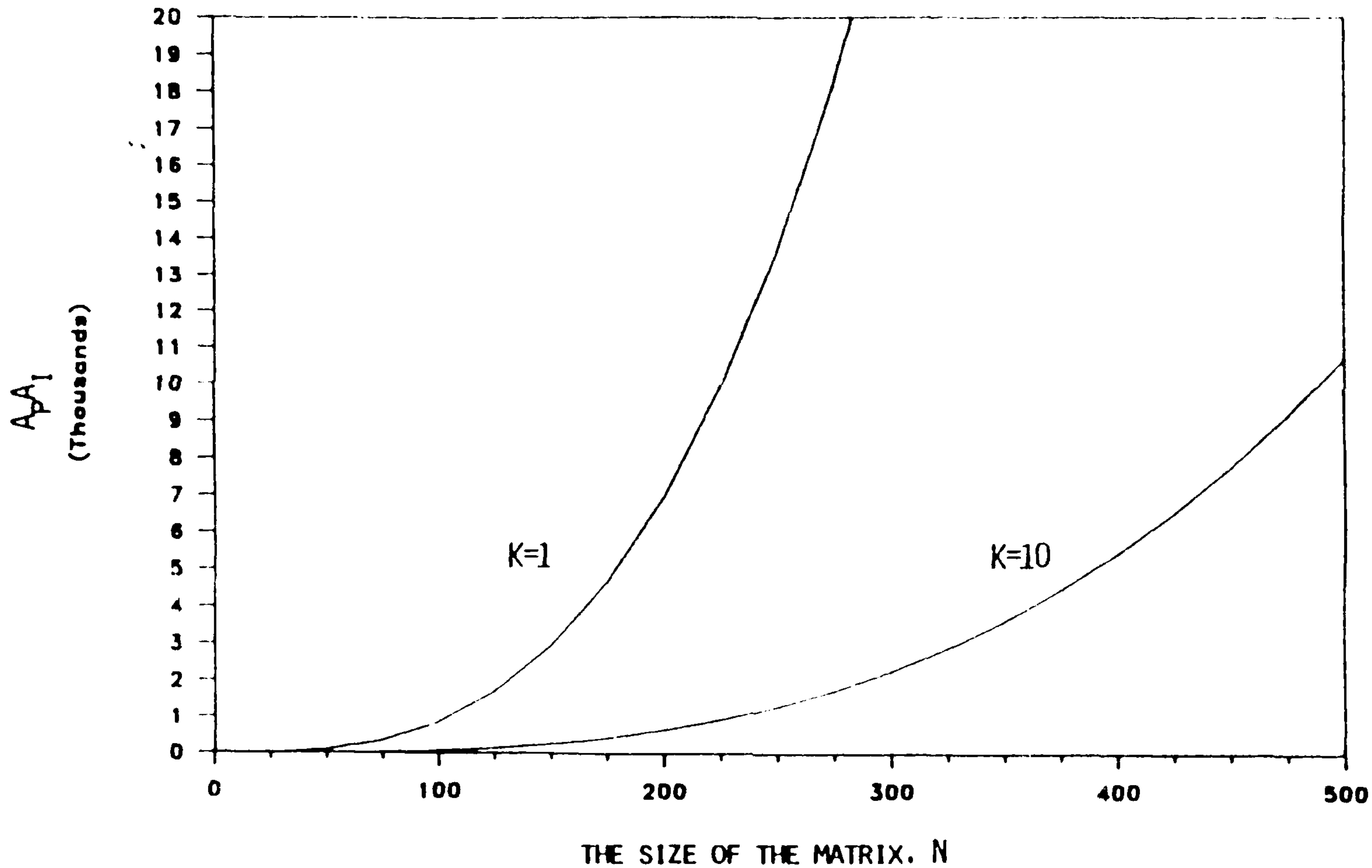


Fig. 8 The factor A_pA_I is plotted as a function of the size of the matrix, N, for K=1 and 10

mask. The transmitted light is summed row wise by using another set of planner waveguides and detected by photodiodes which represent the output vector $\vec{b}$.

The electronic circuit acts as a feedback loop to correct for the input light of the LED's, until a solution is reached. The solution $\vec{x}$ will then be read and stored by the digital processor. Figure 8 shows the electronic circuit used for the feedback loop.

The A/D and D/A conversion from and to the electronic circuit are performed by the digital processor.

In this section we present the experimental results for solving a system of linear equations $A\vec{x} = \vec{b}$ using the BOC, where $A, \vec{b}$, and $\vec{x}$ are all positive.

The Log of the error and that of the residue are plotted versus the number of iterations. The error and the residue are defined as,

$$\text{Error} = \|(\vec{x}-\vec{x}^k)\|/\|\vec{x}\| \tag{30}$$

$$\text{Residue}=\|\vec{r}^k\| \tag{31}$$

Where $\|.\|$, is the Enclidean norm, $\vec{x}$ is the exact solution, $\vec{x}^k$ is the k^{th} iteration solution, and $\vec{r}^k$ is the k^{th} iteration residue.

Since we are dealing only with positive numbers in this paper, we used the absolute value of $\vec{r}$ to solve Eq.(2), then we set:

$$\vec{x}^{(n+1)} = \vec{x}^{(n)} + \Delta\vec{x} \tag{32}$$

when all the components of $\vec{r}$ are positive. And

$$\vec{x}^{(n+1)} = \vec{x}^{(n)} - \Delta\vec{x} \tag{33}$$

if all the components of $\vec{r}$ are negative. We reject the iteration when the components of $\vec{r}$ have different signs and take the previous one. By rejecting some iterations we are actually rejecting some corrections which will slow the convergence process.

In all experiments performed, the iteration process is stopped when a 16 bit accuracy is reached. Fig 9 shows that the BOC started almost 30% error and it needed 6

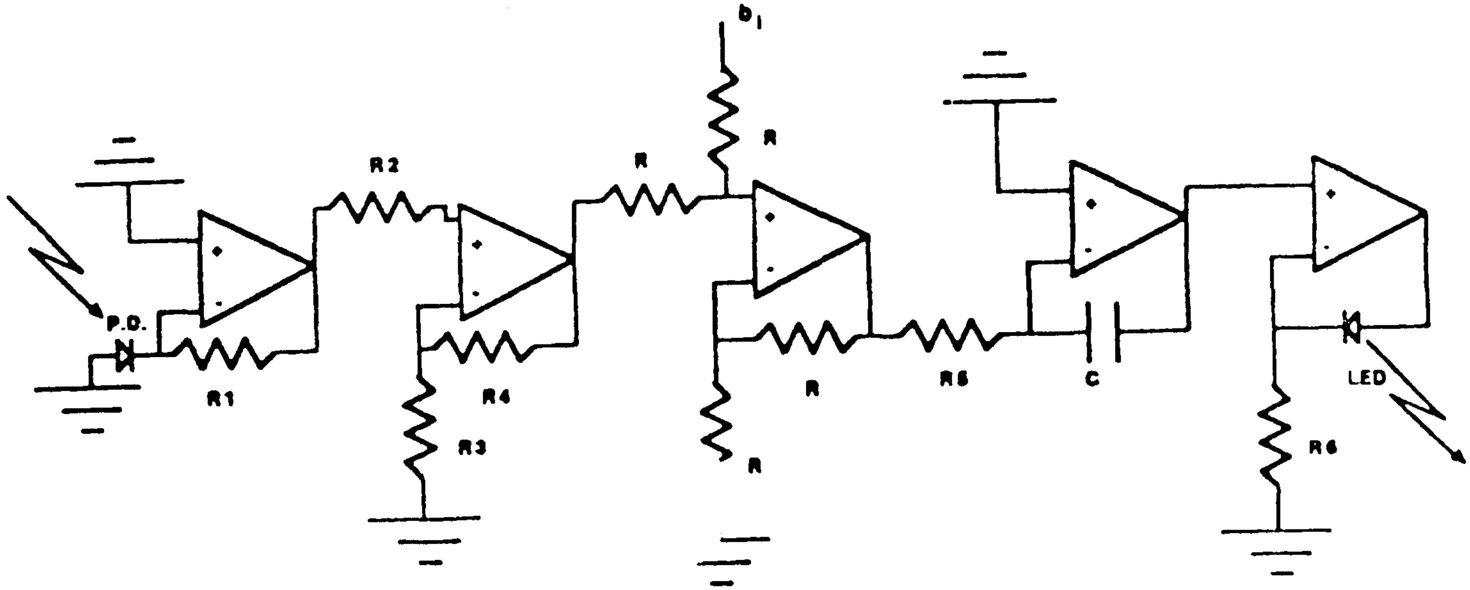


Figure 8 The electronic feedback circuit

iterations to converge to 16 bit accuracy. In Fig 10(a) BOC started with almost 110% error, and the number of iterations needed was 21. Fig 10(b) shows the Log of the residue as a function of the number of iterations. The fluctuations depicted by Figs. 10(a) and (b) is due to the rejection method used in the experiments.

4.1 Effect of Calibration

The analog optical system error is a major factor in the rate of convergence of the BOC. If that error is reduced, then the convergence is much faster. In order to illustrate this, the same problem has been solved twice with two different accuracies of the optical system. The analog optical system's error in the first time was 50%, and it was 30% in the second time. Twenty one iterations were needed by BOC to converge to the 16 bit accuracy for the first case. For the second case the number of iterations was reduced to 12. These results are plotted in Fig 11.

4.2 Reliability of the System

System reliability for convergence have been tested and verified by solving the same problem several times, under different conditions. Results show that when the BOC is used, to solve a problem several times, the convergence rate will not be exactly the same for all the cases. However, the number of iterations needed by the BOC to converge to a certain accuracy is almost the same. Fig 12 shows three different paths of convergence for the same problem. The BOC needed 13 iterations in the first run, 14 iterations in the second, and 11 in the third.

4.3 Convergence of the Singular Matrix

Solving a system of linear equations with a singular matrix A is one of the problems that cannot be solved using conventional digital computer techniques. Singular matrices have a condition number that is equal to infinity, so their inverse does not exist, also they

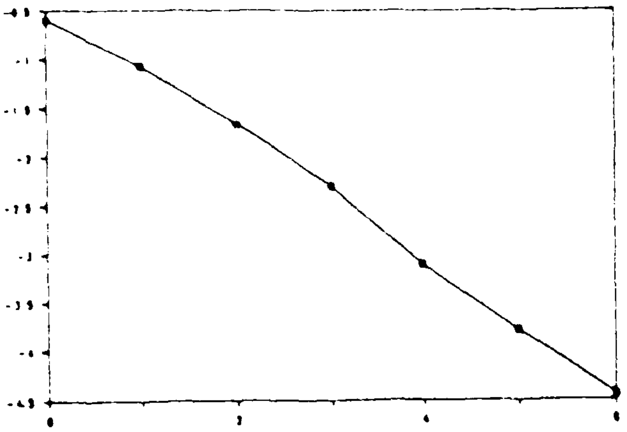


Fig 9 The Log(error) as a function of the number of iterations. The BOC started with 30% error.

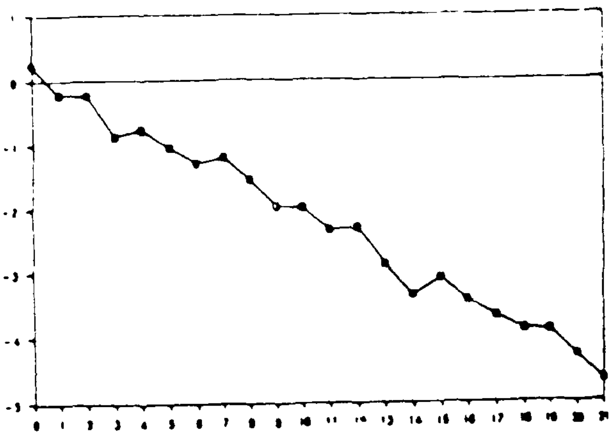


Fig 10(a) The Log(error) as a function of the number of iterations. The BOC started with 100% error.

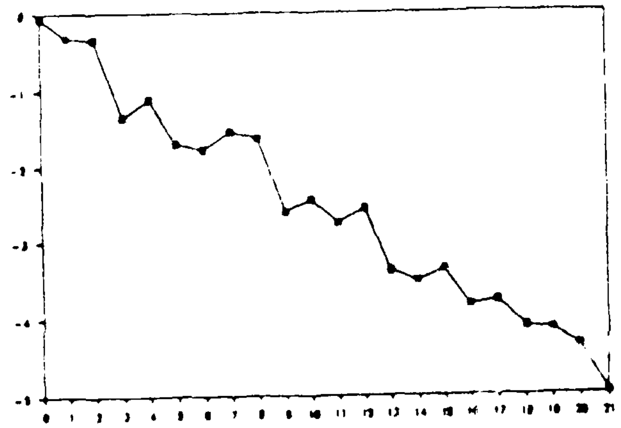


Fig 10(b) The Log(residue) as a function of the number of iterations.

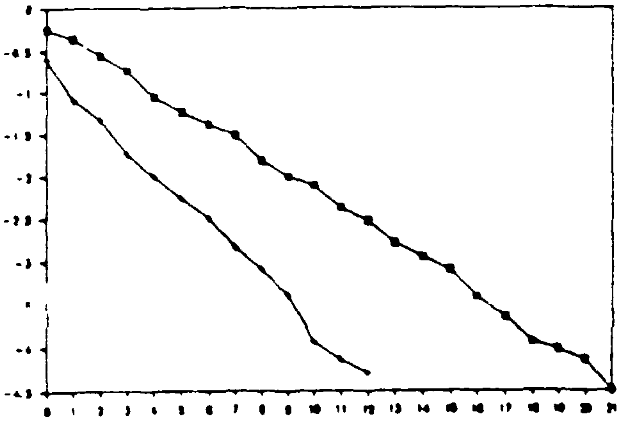


Fig 11 The Log(error) as a function of the number of iterations for the same problem, but with two different accuracies of the optical system.

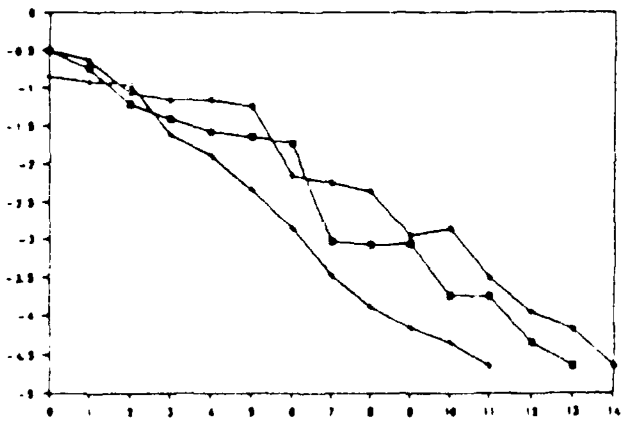


Fig 12 .The Log(error) as a function of the number of iterations for the same problem done three different times.

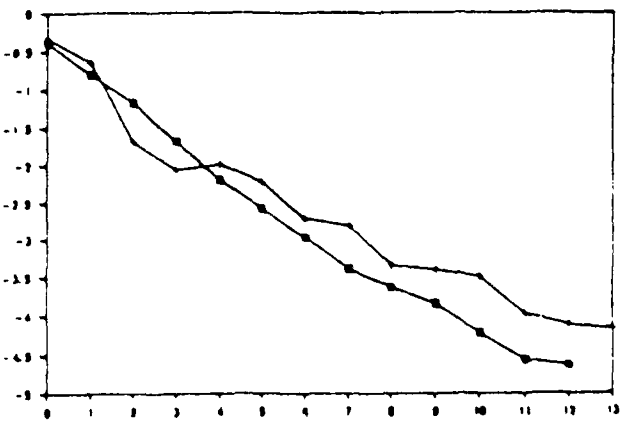


Fig 13 The Log(error) as a function of the number of iterations .The same problem done twice for a singular matrix A.

have infinite solutions. However, the BOC can be used to solve such systems [9]. The BOC converges much faster when A is singular, because a nonsingular matrix will have a unique solution. Due to the infinite solutions that a singular matrix has, the BOC produces different solution each time we try to solve the same problem again. Fig 13 shows the BOC convergence for a singular matrix.

V. Using the BOC for Phased Arrays

It is shown the previous sections that the BOC is capable of large ill–conditioned linear systems of equations very rapdily. This makes the BOC a unique system in processing of phased array data. In this section we show some simulation results using the BOC for the weight adaptation of a phased array.

Two simulation experiments are presented in this section. In the first experiment we used a five element array, and assumed the directions of the jammers are known. In the second experiment a 2 element array is used and no *a priori* information is assumed.

In Fig. 14 the 5 element array pattern is plotted as a function of the angle, ψ. Fig. 14(a) shows the array pattern before adaptation. In Fig. 14(b) the pattern after adaptation is shown for a jammer at 45°. The array pattern after adaptation has reformed in such a way to null the jammer signal. In Fig. 14(c) four jammers are considered at 45°, 80°, 120° and 150°, the array pattern is again reformed to null all the jammers signals reception.

In Fig. 15 the BOC was used to solve the adaptation problem assuming no *a priori* information about the interference signals. Fig. 15(a) shows the two–element array pattern before adaptation. In Fig. 15(b) to (d) the pattern is plotted for a single jammer placed at 30°, 45° and 60°, respectively. In all these plots the array adapted to cancel the interference signal in each of the given cases. In all of the above results the jammer signals is considered to be of the same strength as the desired signal, and the convergence of the solution obtained in less than five iterations. Also the condition number of the R_{XX} is between 10^{6} and ∞.

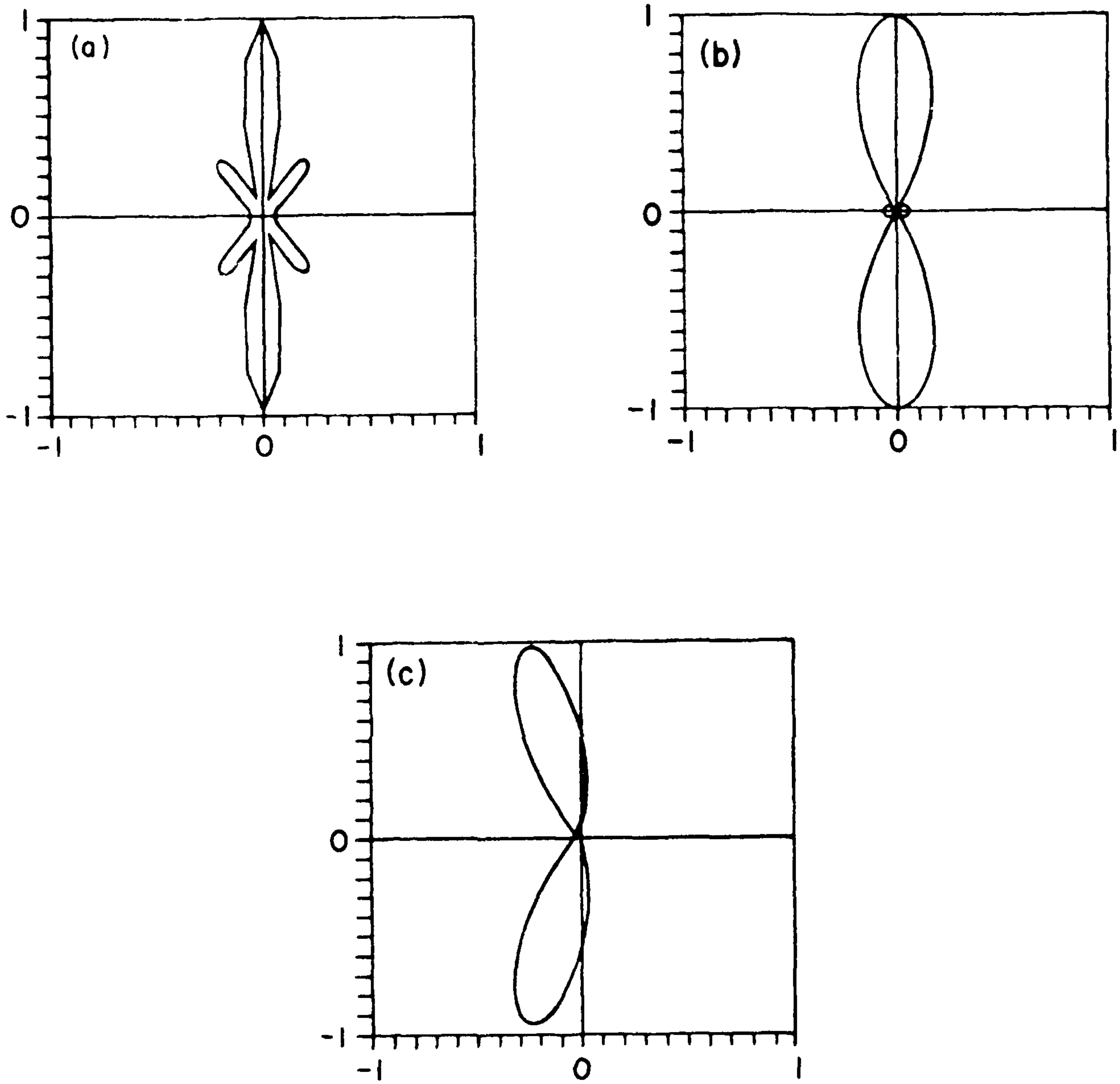


Figure 14 Phased array pattern for 5 elements, (a) before adaptation, (b) adapted pattern for a jammer at 45°, and (c) adapted pattern for four jammers at 45°, 80°, 120° and 150°.

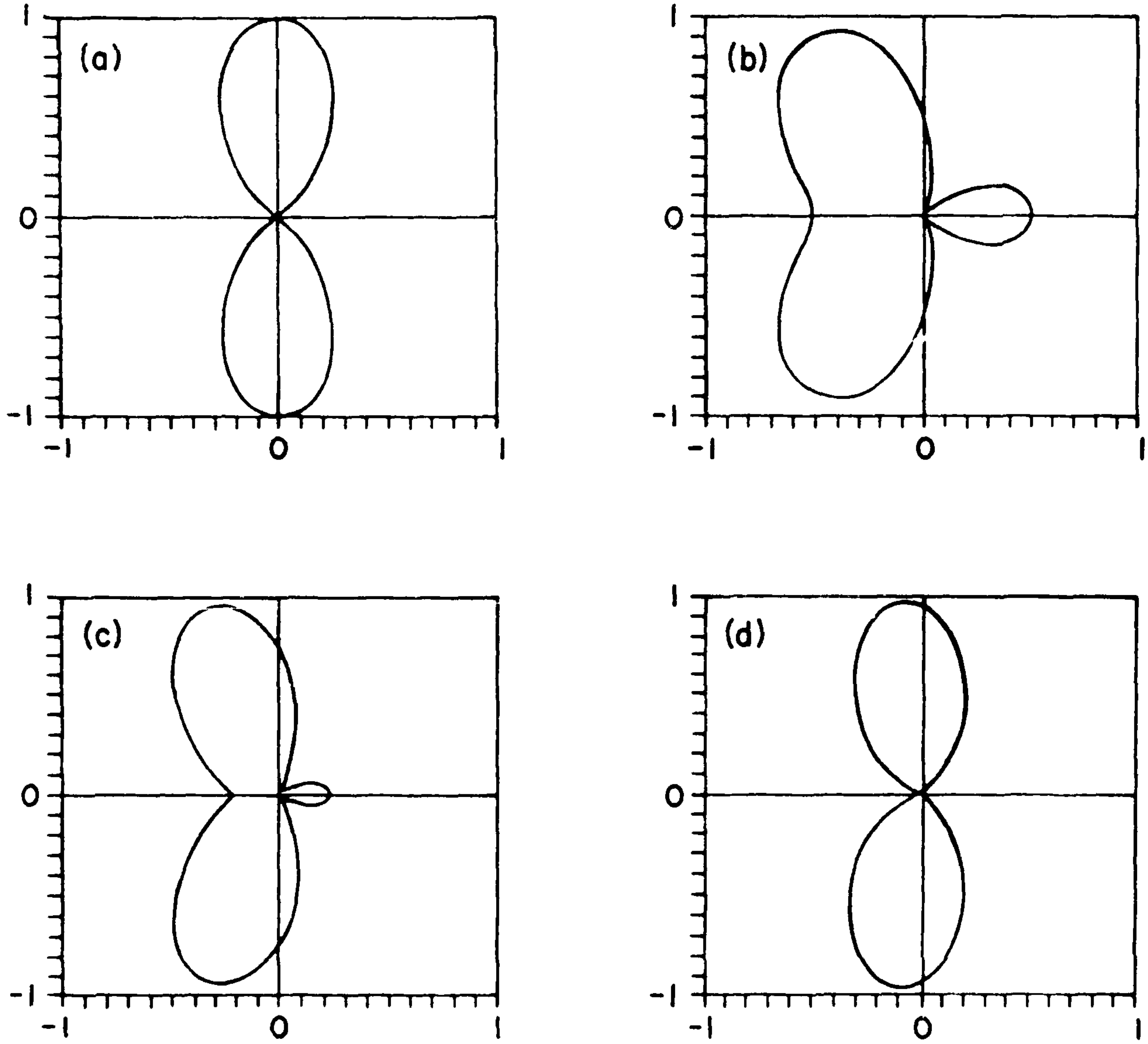


Figure 15 Two element phased array pattern (a) before adaptation, (b) to (d) adapted patterns for single jammers at 30°, 45° and 60°, respectively.

VI. Conclusions

Hybrid optoelectronic processors were demonstrated to be used for processing the adaptive phased arrays data. These processors were shown to have the high accuracy of digital processors while are faster in solving large systems of linear equations. In this particular application of the bimodal optical computer considered here, it has been shown that the BOC is suitable for the adaptation processes of the weights. The interference canceling was achieved even with the presence of the errors in the analog optical processor. Experimental results reported here confirms the theoretical predictions.

For future work the BOC can be implemented to do real time adaptation of the weights. Also other algorithm adaptation can be considered.

REFERENCES

1. B. Widrow, P. E. Mantey, L. J. Griffiths and B. B. Goode, "Adaptive Antenna Systems," Proc. IEEE, 55, p. 2143 (1967).

2. Robert M. Monzingo and Thomas W. Miller, Introduction to Adaptive Arrays, John Wiley and Sons, New York (1980).

3. Bernard Widrow and Samuel D. Stearns, Adaptive Signal Processing, Prentice–Hall, Englewood Cliffs, New Jersey (1985).

4. S. Haykin, Editor, Array Signal Processing, Prentice–Hall, Englewood Cliffs, New Jersey (1985).

5. R. T. Compton, Jr., Adaptive Antennas; Concepts and Performance, Prentice Hall, Englewood Cliffs, New Jersey (1988).

6. H. J. Caulfield, J. H. Gruniger, J. E. Ludman, K. Steiglitz, H. Rabitz, J. Gelfand, and E. Tsoni, "Bimodal Optical Computers," Appl. Opt. 25, 3123 (1986).

7. W. K. Cheng and H. J. Caulfield, "Fully parallel relaxation algebraic operations for optical computers," Opt. Commun. 43, 251 (1982).

8. M. A. G. Abushagur and H. J. Caulfield, "Speed and Convergence of Bimodal Optical Computers," Opt. Eng. 26, 022 (1987).

9. M. A. G. Abushagur, H. J. Caulfield, P. M. Gibson, and M. Habli, "Superconvergence of Hybrid Optoelectronic Processors," Appl. Opt. 26, 4906 (1987).

APPENDICES

The following papers are published during the period of this contract.

1. Mustafa A.G. Abushagur and H. John Caulfield, "Hybrid Optoelectronic Nonlinear Algebra Processor," Dennis Paper, ed., Proc. SPIE 936, 309–314 (1988).

2. M. A. Habli, M.A.G. Abushagur, and H. J. Caulfield, "Solving System of Linear Equations Using the Bimodal Optical Computer; Experimental Results," Dennis Pape, ed., Proc. SPIE 936, 315–320 (1988).

3. Mustafa A.G. Abushagur and H. John Caulfield, "Solving Ill–Posed Algebra Problems Using the Bimodal Optical Computer, "D.P. Casasent and A.G. Tescher, ed., Proc. SPIE 939, 29–33 (1988).

4. Mustafa A.G. Abushagur and Mohamad Habli, "Error Effects on the Processing of Adaptive Array Data Using the BOC," presented at the 32nd SPIE Annual Symposium, San Diego, CA, August 14–19, 1988, Proceedings #975.

5. Mustafa A.G. Abushagur, "Adaptive Array Radar Data Processing Using the Bimodal Optical Computer," <u>Microwave and Optical Technology Letter 1</u>, 236–240 (1988).

A copy of each of these papers is included.

HYBRID OPTOELECTRONIC NONLINEAR ALGEBRA PROCESSOR

Mustafa A. G. Abushagur
Electrical and Computer Engineering Department

H. John Caulfield
Center for Applied Optics
University of Alabama in Huntsville
Huntsville, Alabama 35899

ABSTRACT

A novel system for solving systems of nonlinear equations is proposed. Two different algorithms are introduced. A speed analysis of the two different algorithms is presented and compared with the speed of their digital computer counter parts. A great advantage in speed is shown for large size problems.

1. INTRODUCTION

Systems of nonlinear equations arise in the process of solving many physical problems. They are a very important class of mathematical problems. Iterative methods are used to solve such problems.

In this paper we propose a new method for solving this class of nonlinear problems using optical processors. In Section 2 the iterative methods used in solving nonlinear systems of equation is reviewed. In Section 3 the optical implementation is proposed using two different algorithms. The speed analysis of the two algorithms is given in Section 4. In Section 5 conclusions and final remarks are drawn.

2. NEWTON's METHOD

Systems of linear equations are given as follows

$$A\vec{x} = \vec{b} \tag{1}$$

where A is an $n \times n$ matrix, $\vec{x}$ and $\vec{b}$ are $n \times 1$ vectors. In these systems A and $\vec{b}$ are given and $\vec{x}$ the solution of the system is unknown.

Nonlinear systems of equations can be represented by

$$\vec{f}(\vec{x}) = 0 \tag{2}$$

or

$$\begin{bmatrix} f_1(x_1 \cdots x_n) \\ \cdots\cdots\cdots \\ f_n(x_1 \cdots x_n) \end{bmatrix} = 0,$$

where f_i's are nonlinear functions of $\vec{x}$

One of the methods used in solving for $\vec{x}$ in the nonlinear system of equations is Newton's method. For a single nonlinear equation, an initial solution, x_o, of the equation is assumed, and the $(k+1)^{th}$ iteration of the solution is given by[1]

$$x_{k+1} = x_k - (f'_k)^{-1} f_k \tag{3}$$

where

$$f_k = f(x_k) \text{ , and } f'_k = \left.\frac{\partial f(x)}{\partial x}\right|_{x = x_k} \tag{4}$$

For a system of nonlinear equations, Eq. (3) can be rewritten as

$$\vec{x}_{k+1} = \vec{x}_k - (J_k)^{-1} \vec{f}_k \ , \tag{5}$$

where

$$(J)_{i,j} = \frac{\partial f_i(\vec{x})}{\partial x_j} \ , \tag{6}$$

and J is the Jacobian matrix.

Let

$$(J_k)^{-1} \vec{f}_k = \vec{c}_k \ , \tag{7}$$

then

$$J_k \vec{c}_k = \vec{f}_k \ . \tag{8}$$

Eq. (8) is a system of linear equations to be solved for $\vec{c}_k$, which is the correction needed for the $(k+1)^{th}$ solution iteration. The algorithm for solving the system of nonlinear equations will be as follows:

i) Assume a solution $\vec{x}_o$.

ii) Compute the nx1 vector $\vec{f}_k$ and the nxn matrix J_k

iii) Solve the linear system of equations $J_k \vec{c}_k = \vec{f}_k$ for $\vec{c}_k$

iv) Compute the refined solution $\vec{x}_{k+1} = \vec{x}_k - \vec{c}_k$

v) Check if the norm $\|\vec{f}_{k+1} - \vec{f}_k\| < \epsilon$ stop, otherwise go back to step (ii). ϵ is the allowable error.

3. OPTICAL IMPLEMENTATION

The iterative algorithm introduced in Section 2 requires $0(n^3)$ number of operations when used with conventional digital computer. The most expensive part of the algorithm is step (iii) to solve a system of linear equations. In previous publications[2-4] we have proposed and analyzed a hybrid optoelectronic processor, the Bimodal Optical Computer BOC, capable of solving linear systems of equations accurately and rapidly. In this section we modify that system to be used to solve systems of nonlinear equations as shown in Fig. 1. We propose two different algorithms, the first utilizes the use of the analog processor to solve the system of equations (8) approximately, and the second to use the BOC to solve the system of equations (8) exactly (within the specified accuracy).

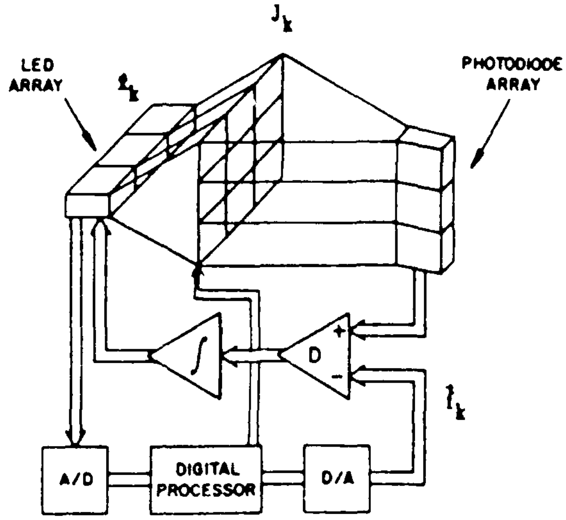


Fig.1 Block diagram of the hybrid optoelectronic system.

3.1 Hybrid Analog Optical Processor

In this system we use the optical analog processor to solve Eq. (8) approximately. For this system we introduce the following algorithm:

a) Use the digital processor to guess an initial solution $\vec{x}_o$.

b) Use the digital processor to compute both the vector $\vec{f}_k$ and the matrix J_k.

c) Use the optical analog processor to solve the system $J_k^o \vec{c}_k^o = \vec{f}_k^o$ for $\vec{c}_k^o$, approximately, where the superscript o's denote inaccuracies in optics or electronics.

d) Use the digital processor to read $\vec{c}_k^o$ and compute the refined solution $\vec{x}_{k+1} = \vec{x}_k - \vec{c}_k^o$.

e) Check if the norm $\|\vec{f}_{k+1} - \vec{f}_k\| < \epsilon$ stop, otherwise, go back to step (b) and recycle.

3.2 Hybrid BOC Processor

In this system the BOC is used to solve Eq.(8) exactly. For this system we introduce the following algorithm:

a) Use the digital processor to guess an initial solution $\vec{x}_o$.

b) Use the digital processor to compute both $\vec{f}_k$ and the matrix J_k.

c) Use the BOC to solve the system $J_k \vec{c}_k = \vec{f}_k$, exactly for $\vec{c}_k$.

d) Use the digital processor to read $\vec{c}_k$ and compute the refined solution $\vec{x}_{k+1} = \vec{x}_k - \vec{c}_k$.

e) Check if the norm $\|\vec{f}_{k+1} - \vec{f}_k\| < \epsilon$ stop, otherwise, go back to step (b) and recycle.

4. SPEED ANALYSIS

The following speed analysis is based on a system of linear equations with size, n.

4.1 Digital Processor

The total time required, T_{DT}, to solve the system of nonlinear equations using a conventional digital processor is given by

$$T_{DT} = [\frac{n^3}{3} + 2n(n+1)]T_{D1}N_D \tag{9}$$

where

T_{D1} = the time needed to do one digital operation (e.g., a multiplication),

and

N_D = the number of iterations needed for the solution convergence.

4.2 Hybrid Analog Optical Processor

The total time required, T_{OA}, to solve the system of nonlinear equations using the processor introduced in Section 3.1 is given by

$$T_{OA} = [n(n+2)T_{D1} + T_{A1}]\, N_A \tag{10}$$

where

T_{A1} = the [illegible] required for the optical analog processor to solve the system of linear equations (8) approximately,

and

N_A - the number of iterations required for the solution convergence.

4.3 Hybrid BOC Processor

The total time required, T_{OB}, to solve the system of nonlinear equations using the processor introduced in Section 3.2 is given by

$$T_{OB} = [2n(n+1)T_{D1} + T_{A1}]\, I_B N_D \tag{11}$$

where

I_B = the number of iteration needed for the BOC to solve Eq. (8) to the specified accuracy

4.4 Speed Advantage

It is of great interest to determine what is the break even point for the optical processor proposed to be faster than the digital processors. This condition is defined by

$$T_{DT} > T_{OA} \tag{12}$$

and

$$T_{DT} > T_{OB} \; . \tag{13}$$

From Eqs. (9) to (11) the conditions (12) and (13) can be written as

$$\left(\frac{n^2(n/3+1)}{I_A}\right) \times \left(\frac{T_{D1}}{T_{A1}}\right) > 1 \tag{14}$$

or

$$A_n \times A_t > 1 \tag{15}$$

for the hybrid analog processor, where

$$I_A = N_A/N_D . \tag{16}$$

And for the hybrid BOC processor

$$\left(\frac{n^3/3 - 2n(n+1)(I_B-1)}{I_B}\right) \times \left(\frac{T_{D1}}{T_{A1}}\right) > 1 \; . \tag{17}$$

or

$$B_n \times A_t > 1 \; . \tag{18}$$

Where

$$A_n = \frac{n^2(n/3+1)}{I_A} \; . \tag{19}$$

$$B_n = \frac{n^3/3 - 2n(n+1)(I_B-1)}{I_B} \; . \tag{20}$$

and

$$A_t = \frac{T_{D1}}{T_{A1}} . \tag{21}$$

The number of iterations, I_A and I_B, usually are in the range of 1 to 10.[1] The ratios, A_n and B_n are problem dependent, and are much larger than 1 for large values of n. On the other hand, A_t depends on the speed of the analog processor for solving a system of linear equation, which can be in the range of μsec. But since the matrix J_k need to be updated every cycle, writing the matrix J_k on the SLM becomes the bottleneck of the processor speed. With todays technology writing a matrix on an SLM may take a few milliseconds. So A_t is much less than 1. In Fig. 2(a) and (b) the Log (A_n) and Log (B_n) are plotted in terms of the system size, n, respectively. The ratio $A_n \gg 1$, for $n \simeq 10$, while $A_n \gg 1$ for $n \cong 60$ and 120, for $I_B = 10$ and 20 respectively. For the A_t ratio in the range of 10^{-3}, we can have a speed advantage for the hybrid analog optical processor for $n \geq 50$, and for the hybrid BOC processor for $n \geq 120$.

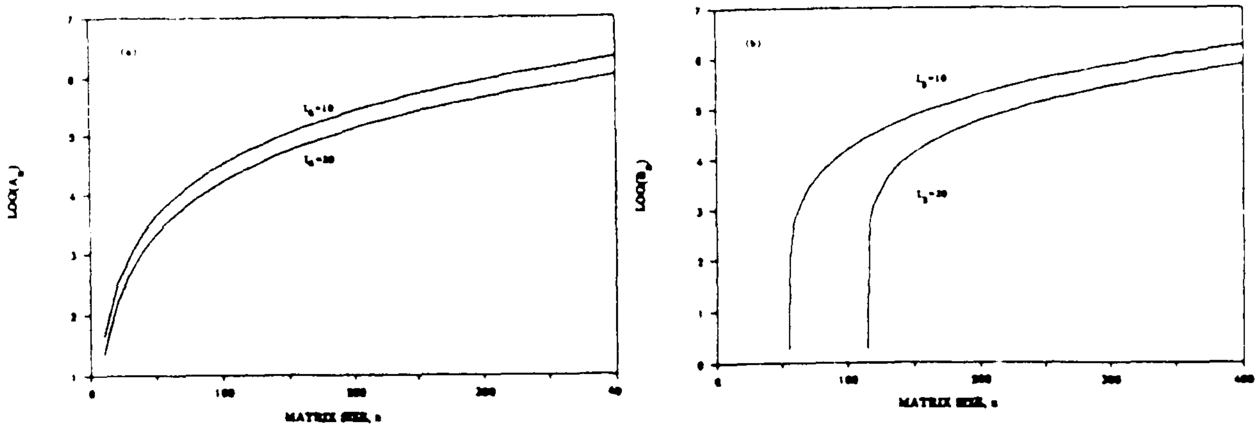


Fig.2 Plot of log of the ratio (a) A_n of Eq.(19), and (b) B_n of Eq.(20), in terms of the size of the matrix ,n.

Again this ratio A_t depends mainly on how fast we can write a matrix on the SLM. By the introduction of faster SLM's the speed advantage can be gained for smaller values of n.

5. CONCLUSIONS

Two new hybrid opto electronic processors are introduced for solving systems of nonlinear equations. The speed of the two processors is analyzed and compared with the speed of digital processors. It is shown that the main factor of the speed limitation is the speed the SLM's used to write the matrix on. With the existing SLM's a speed advantage can be gained for $n \geq 100$.

6. ACKNOWLEDGEMENTS

This research was supported by the Innovative Science and Technology Office of the Strategic Defense Initiative Organization, administered through the Office of Naval Research under contract N00014-86-k-0591, and by Rome Air Development Center, administered through Georgia Institute of Technology under contract E-21-669-S23.

REFERENCES

1. R. L. Johnson, Numerical Methods, John Wiley, New York (1982).
2. H. J. Caulfield, J. H. Gruniger, J. E. Ludman, K. Steiglitz, H. Rabitz, J. Gelfand, and E. Tsoni, "Bimodal Optical Computers," Appl. Opt. 25, 3123 (1986).
3. M. A. G. Abushagur and H. J. Caulfield, "Speed and Convergence of Bimodal Optical Computers," Opt. Eng. 26, 022 (1987).
4. M. A. G. Abushagur, H. J. Caulfield, P. M. Gibson and M. Habli, "Superconvergence of Hybrid Optoelectronic Processors," App. Oct. 26, 4906 (1987).

SOLVING SYSTEM OF LINEAR EQUATIONS USING THE BIMODAL OPTICAL COMPUTER (EXPERIMENTAL RESULTS)

M. A. Habli
M. A. G. Abushagur
Electrical and Computer Engineering Department
University of Alabama in Huntsville
Huntsville, Alabama 35899

H. J. Caulfield
Center for Applied Optics
University of Alabama in Huntsville
Huntsville, Alabama 35899

ABSTRACT

Hardware and software design of the Bimodal Optical Computer (BOC) and its implementations are presented. Experimental results of the BOC for solving a system of linear equations $A\underline{x} = \underline{b}$ is reported. The effect of calibration, the convergence reliability of the BOC, and the convergence of problems with singular matrices are studied.

1.INTRODUCTION

Analog optical systems are becoming very attractive in the area of signal processing because of their ability to process in parallel two dimensional data very rapidly. However, analog optical systems have low accuracy. BOC [1–4] solves this low accuracy problem, by using a combination of both analog optical system and digital processor.

In this paper we present experimental results using BOC for solving systems of linear equations. In Section 2 a comparison between astigmatic optics and waveguides based algebra processors is presented. The hardware and the software design of BOC is in Section 3. Section 4 contains the experimental results of the BOC for solving a system of linear equations. The conclusions are in Section 5.

2.ASTIGMATIC OPTICS AND WAVEGUIDES BASED ALGEBRA PROCESSORS

The analog optical system can be applied in many applications. This paper concentrates on solving a system of linear equations. Goodman [5] has introduced an astigmatic processor to perform matrix vector multiplications, which can also be used in a system of linear equations solver. However, the main problem that faces the arrangement in Fig. 1 is aligning the components, to insure a uniform light distribution along the matrix plane.

Waveguides can be used to build optical algebra processors. By using waveguides, the optical system can be made compact, and its alignment will be much easier than that of the astigmatic system. The distribution of the light across the waveguide is plotted in Fig. 2, which shows that the light is almost uniform along the waveguide. From the practical standpoint waveguides are more reliable to use in these systems than the astigmatic optics.

3.THE BOC DESIGN (HARDWARE AND SOFTWARE)

3.1 BOC HARDWARE DESIGN

The BOC hardware has three main parts as shown in Fig. 3. The optical system, the electronic circuit, and the digital processor. The optical system consists of the fully parallel matrix–vector multiplier. Light from the LED's representing the $\underline{x}$ components are spread vertically by planner waveguides onto the columns of the matrix mask. The transmitted light is summed row wise by using another set of planner waveguides and detected by photodiodes which represent the output vector $\underline{b}$.

The electronic circuit acts as a feedback loop to correct for the input light of the LED's, until a solution is reached. The solution $\underline{x}$ will then be read and stored by the digital processor. Fig. 4 shows the electronic circuit used for the feedback loop.

The A/D and D/A conversion from and to the electronic circuit are performed by the digital processor.

3.2 BOC SOFTWARE DESIGN

The BOC software controls the Input/Output operations. Both the matrix A and the output vector $\underline{b}$ are read and stored by the digital processor. The vector $\underline{b}$ is then converted to analog voltage by a D/A converter, and it is assigned to the different ports of the electronic circuit. The analog optical processor solves for an approximate solution due to its inaccuracy. The digital processor reads and stores the approximate solution, $\underline{x}^o$ through the A/D converter, then it calculates the residue vector, $\underline{r}$, as,

$$\underline{r}=\underline{b}-A\underline{x}^{o}=A(\underline{x}-\underline{x}^{o})=A\Delta\underline{x} \tag{1}$$

Multiply Eq. (1) by a scalar s to make use of the whole dynamic range of the system, so Eq (1) becomes,

$$s\underline{r}=A(s\Delta\underline{x}) \tag{2}$$

If the residue is not small enough, the system of linear Eq.(2) will be solved for $\Delta\underline{x}$ using the analog optical processor and,

$$\underline{x}^{1}=\underline{x}^{o}+\Delta\underline{x} \tag{3}$$

A new residue will be found for $\underline{x}^1$. The iteration process is continued by solving Eqs.(1) through (3) until a satisfactory solution is reached.

4.EXPERIMENTAL RESULTS

In this section we present the experimental results for solving a system of linear equations $A\underline{x}=\underline{b}$ using the BOC, where A,$\underline{b}$, and $\underline{x}$ are all positive.

The Log of the error and that of the residue are plotted versus the number of iterations. The error and the residue are defined as,

$$\text{Error}=||(\underline{x}-\underline{x}^{k})||/||\underline{x}|| \tag{4}$$

$$\text{Residue}=\|\underline{r}^k\| \tag{5}$$

Where $\|.\|$, is the Enclidean norm, $\underline{x}$ is the exact solution, $\underline{x}^k$ is the k^{th} iteration solution, and $\underline{r}^k$ is the k^{th} iteration residue.

Since we are dealing only with positive numbers in this paper, we used the absolute value of $\underline{r}$ to solve Eq.(2), then we set:

$$\underline{x}^{(n+1)}=\underline{x}^{(n)}+\Delta\underline{x} \tag{6}$$

when all the components of $\underline{r}$ are positive. And

$$\underline{x}^{(n+1)}=\underline{x}^{(n)}-\Delta\underline{x} \tag{7}$$

if all the components of $\underline{r}$ are negative. We reject the iteration when the components of $\underline{r}$ have different signs and take the previous one. By rejecting some iterations we are actually rejecting some corrections. This procedure slows down the convergence process.

In all the experiments performed, the iteration process is stopped when a 16 bit accuracy is reached. Fig. 5 shows that the BOC started with almost 30% error and it needed 6 iterations to converge to 16 bit accuracy. In Fig. 6(a) BOC started with almost 110% error, and the number of iterations needed was 21. Fig. 6(b) shows the Log of the residue as a function of the number of iterations. The fluctuations depicted by Figs. 6(a) and (b) is due to the rejection method used in the experiments.

4.1 EFFECT OF CALIBRATION

The analog optical system error is a major factor in the rate of convergence of the BOC. If that error is reduced, then the convergence is much faster. In order to illustrate this, the same problem has been solved twice with two different accuracies of the optical system. The analog optical system's error in the first time was 50%, and it was 30% in the second time. Twenty one iterations were needed by BOC to converge to the 16 bit accuracy for the first case. For the second case the number of iterations was reduced to 12. These results are plotted in Fig. 7.

4.2 RELIABILITY OF THE SYSTEM

System reliability for convergence have been tested and verified by solving the same problem several times, under different conditions. Results show that when the BOC is used, to solve a problem several times, the convergence rate will not be exactly the same for all the cases. However, the number of iterations needed by the BOC to converge to a certain accuracy is almost the same. Fig. 8 shows three different paths of convergence for the same problem. The BOC needed 13 iterations in the first run, 14 iterations in the second, and 11 in the third.

4.3 SOLUTION CONVERGENCE FOR THE SINGULAR MATRIX SYSTEM

Solving a system of linear equations with a singular matrix A is one of the problems that cannot be solved using conventional digital computer techniques. Singular matrices have a condition number equal to infinity, so their inverse does not exist, also they have infinite number of solutions. However, the BOC can be used to solve such systems [6]. The BOC converges much faster when A is singular, because a nonsingular matrix will have a unique solution. Due to the infinite solutions that a singular matrix has, the BOC produces different solution each time we try to solve the same problem again. Fig. 9 shows the BOC convergence for a singular matrix.

5.CONCLUSIONS

The BOC system was built and experimentally tested. The experimental results show great reliability of the processor in solving systems of linear equations. Overall 16 bit accuracy of the hybrid system was achieved with an analog optical system of 30% to 50% error. Higher accuracies of the solution can be obtained by increasing the number of iterations. The BOC also demonstrated to solve systems of linear equations with singular matrices.

We are considering in future work, bipolar numbers, complex numbers, and using SLM for the matrix mask.

6.ACKNOWLEDGEMENTS

This research was supported by the Innovative Science and Technology Office of the Strategic Defense Initiative Organization, administered through the Office of Naval Research under contract N00014-86-k-0591, and by Rome Air Development Center, administered through Georgia Institute of Technology under contract E-21-669-S23.

REFERENCES

1. H.J.Caulfield, J.H.Gruninger, J.E.Ludman, K.Steiglitz, H.Rabitz, J.Gelfand, and E.Tsoni, "Bimodal optical computers", Appl. Opt. 25(18),3123-3131, (1987).

2. W.K.Cheng and H.J.Caulfield, "Fully parallel relaxation algebraic operations for optical computers," Opt. Commun. 43,251 (1982).

3. M.A.G.Abushagur, and H.J.Caulfield, "Speed and convergence of the Bimodal optical computers", Opt. Eng. 26(1), 022-027 (1987).

4. M.A.G.Abushagur, H.J.Caulfield, and M.A.Habli, "Computer simulation of the Bimodal Optical Computer", Southeastern Computer Simulation Conference, SESC '87.

5. J.W.Goodman, A.R.Dias, and L.M.Woody, "Fully parallel, high-speed incoherent optical method for performing discrete Fourier transforms, "Opt. Lett. 2, 1 (1978).

6. M.A.G.Abushagur, H.J.Caulfield, P.M.Gibson, and M.A.Habli, "Superconvergence of hybrid optoelectronic processor", App. Opt. , Vol. 26, (23), 4906, 1987.

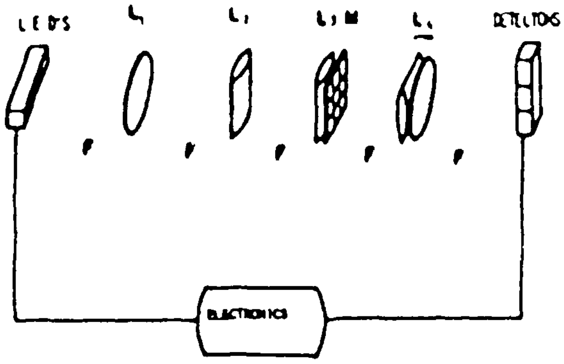


Fig 1. An astigmatic processor for solving system of linear equations.

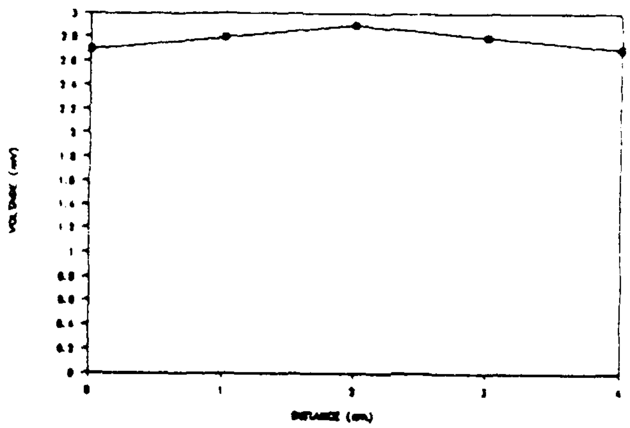


Fig 2. Light distribution across the waveguide.

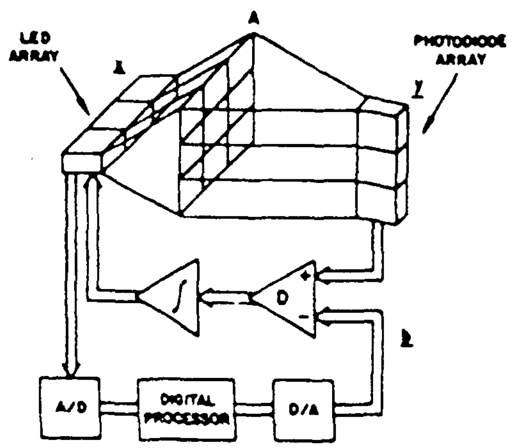


Fig 3. The Bimodal Optical Computer.

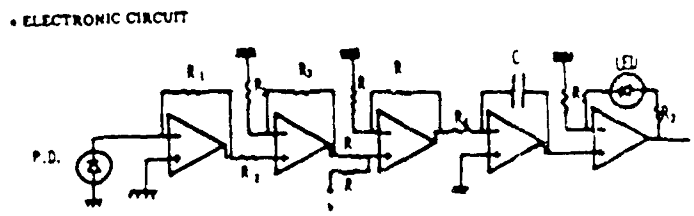


Fig 4. circuit diagram of the feedback loop for the optical system.

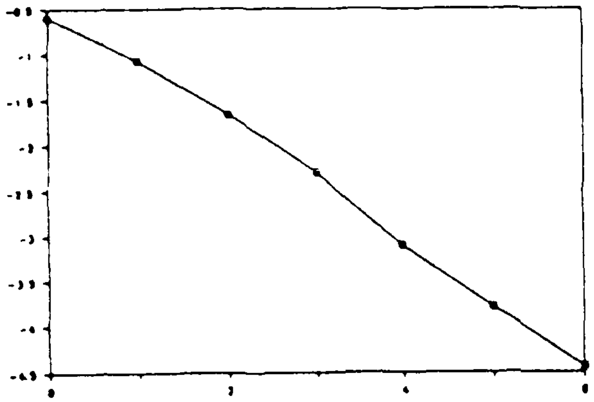


Fig 5. The Log(error) as a function of the number of iterations. The BOC started with 30% error.

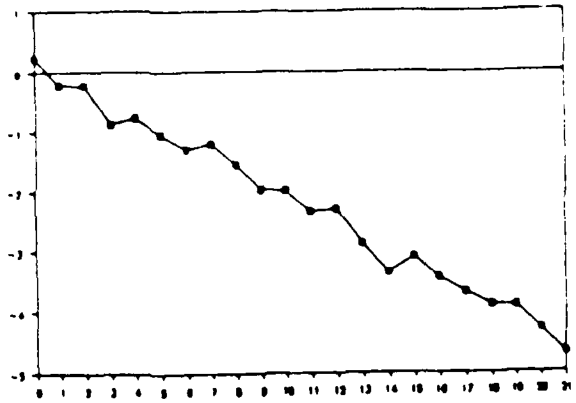


Fig 6(a). The Log(error) as a function of the number of iterations. The BOC started with 100% error.

Fig 6(b). The Log(residue) as a function of the number of iterations.

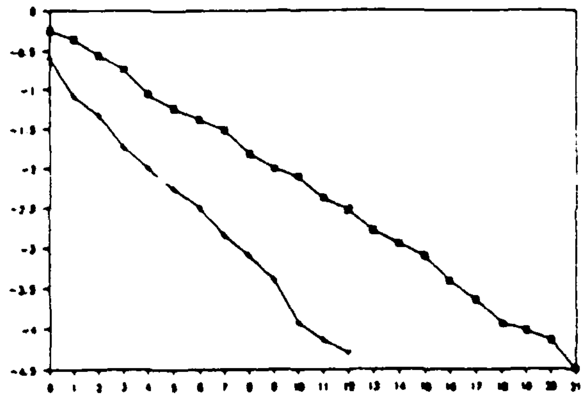


Fig 7. The Log(error) as a function of the number of iterations for the same problem, but with two different accuracies of the optical system.

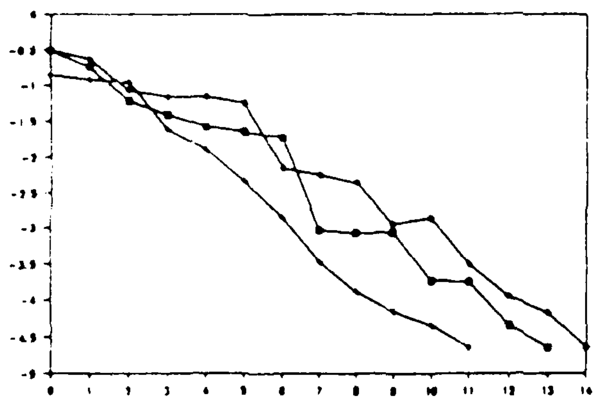


Fig 8. The Log(error) as a function of the number of iterations for the same problem done three different times.

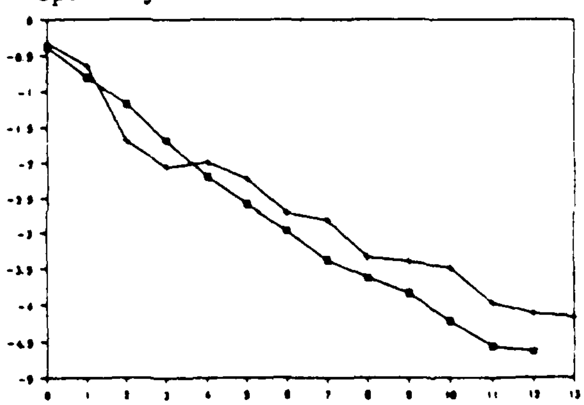


Fig 9. The Log(error) as a function of the number of iterations .The same problem done twice for a singular matrix A.

SOLVING ILL–POSED ALGEBRA PROBLEMS USING THE BIMODAL OPTICAL COMPUTER

Mustafa A. G. Abushagur
Electrical and Computer Engineering Department

H. John Caulfield
Center for Applied Optics
University of Alabama in Huntsville
Huntsville, Alabama 35899

ABSTRACT

A set of ill–posed algebra problems are considered for solving by the Bimodal Optical Computer, BOC. The BOC algorithm was shown to be capable of solving this class of algebra problems. Three different methods of generating the error matrix are compared in terms of the convergence of the solution. Some applications for the methods are introduced.

1. INTRODUCTION

Optical linear algebra processors[1-4] are introduced to provide a fast and reliable means of solving linear algebra problems. We have shown in previous publications[5] that the BOC is capable of solving systems of linear equations with singular matrices. The BOC algorithm which we have introduced can also be used on conventional digital computers. In this paper we present three different methods in generating the error matrix used in the algorithm. A comparison between the different methods is shown in terms of the solution convergence.

In Section 2 we review the BOC algorithm for solving a system of linear equations. Different schemes in generating the error matrix is given in Section 3. Application of the method to semilinear problems is shown in Section 4. In Section 5 conclusions and final remarks are given.

2. THE BOC ALGORITHM

In this section we review the BOC algorithm to solve a system of linear equations. The algorithm originally was developed for the Bimodal Optical Computer, BOC, which is a hybrid system, combining both analog optics and digital electronics to achieve accurate solutions very rapidly for systems of linear equations. The algorithm can also be used on a digital computer to solve systems with very ill–conditioned or singular matrices.

The problem we are interested to solve is the basic system of linear equations

$$A\vec{x} = \vec{b} \ . \qquad (1)$$

where A is an $n \times n$ matrix, $\vec{x}$ and $\vec{b}$ are $n \times 1$ vectors. A and $\vec{b}$ are are known, and $\vec{x}$ is the solution of the system we need to determine. The matrix A can be either an ill–conditioned or singular.

The BOC algorithm is as follows:

(a) Generate the matrix A^o, which is given by

$$A^o = A + E , \tag{2}$$

where E is the error matrix.

$$E = \begin{bmatrix} \epsilon_{11} & \cdots & \epsilon_{1n} \\ \cdot & \cdot & \cdot \\ \epsilon_{i1} & \cdots & \epsilon_{in} \\ \cdot & \cdot & \cdot \\ \epsilon_{n1} & \cdots & \epsilon_{nn} \end{bmatrix} , \tag{3}$$

and is generated using a Gaussian random number generator, with zero mean and standard deviation, σ_E.

(b) Solve the system

$$\vec{A}^o \vec{x}^o = \vec{b} , \tag{4}$$

for $\vec{x}^o$.

(c) Compute the residue

$$\vec{r} = \vec{b} - A\vec{x} = A \, \Delta\vec{x} . \tag{5}$$

(d) Solve the system

$$s\vec{r} = A^o \, s\Delta\vec{x}^o \tag{6}$$

$\Delta\vec{x}^o$, where s is a constant used to utilize the dynamic range of the system, such that

$$s\| \vec{r} \|_\infty = 1 . \tag{7}$$

(e) Compute the refined solution

$$\vec{x}^1 = \vec{x}^o + \Delta\vec{x}^o \tag{8}$$

(f) if $\| \vec{r} \| < \epsilon$, the allowable error, stop, otherwise, go to step (c) and reiterate.

In this algorithm the matrix A^o is a well conditioned matrix[4]. If A is a singular matrix then it has at least one of its eigenvalues is equal to zero. By adding the error matrix E to A then the zero eigenvalues of A will be shifted and the matrix A^o will have no zero eigenvalues. It is also worth mentioning here that step (a) is not necessary when using an analog optical processor, because it is done naturally.

3. GENERATING THE ERROR MATRIX

In this section we introduce other ways of generating the error matrix, E, given in Eq. (2).

Also we will compare these different methods in terms of the solution convergence of system of linear equations with a singular matrix.

The error matrix, E, can be generated as given by Eq. (3). Also some of the other means it can be generated are

$$E_1 = \begin{bmatrix} \epsilon_{11} & & 0 \\ & \ddots \; \epsilon_{ii} \; \ddots & \\ 0 & & \epsilon_{nn} \end{bmatrix} . \tag{9}$$

or

$$E_2 = \begin{bmatrix} \epsilon & & 0 \\ & \ddots \; \epsilon \; \ddots & \\ 0 & & \epsilon \end{bmatrix} . \tag{10}$$

or

$$E_3 = \begin{bmatrix} 1 & & 0 \\ & \ddots \; 1 \; \ddots & \\ 0 & & 1 \end{bmatrix} + C_2 = \begin{bmatrix} 0 & & 1 \\ & \ddots \; 0 \; \ddots & \\ 1 & & 0 \end{bmatrix} . \tag{11}$$

In Eq. (9) the diagonal matrix is generated using Gaussian statistics.

In Fig. 1 the number of iterations required for a 16 bit solution of a 5 × 5 singular system is plotted in terms of the logarithm of the standard deviation, σ_E, of matrix E and E_1 or the logarithm of ϵ of matrix E_2. In Fig. 1(a) a 5 × 5 matrix with rank = 4 is used, while in Fig. 1(b) the rank of the matrix is only 1.

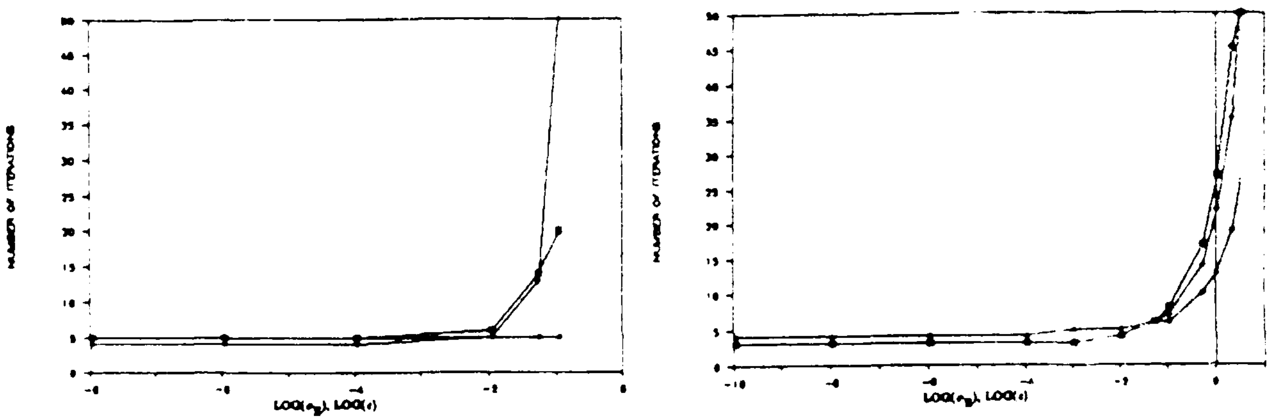
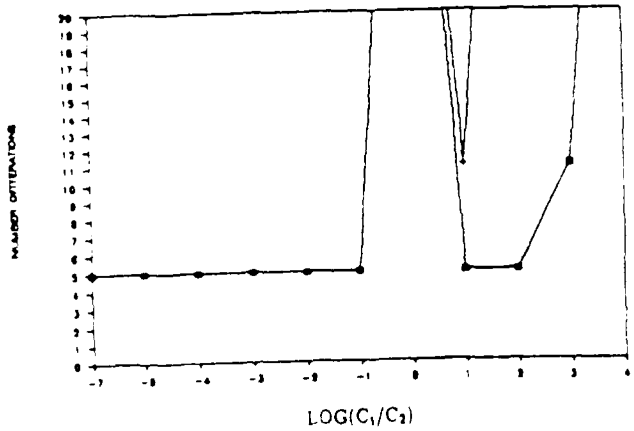


Fig.1 Number of iterations required for 16 bit resolution is plotted in terms of the logarithm of σ_E or ϵ for a 5x5 matrix with (a) rank=4, and (b) rank=1.

In Fig. 1(a) the number of iterations using E and E_1 is 5 for an ernumber of iteration starts to increase. While the number of iterations using E_2 is only 4 for an error, ϵ, up to 10%. For rank 1 matrix, Fig. 1(b), the number of iterations is 3 or 4 for an error up to 10%. When the error reaches 100% the number of iterations is between 15 and 25, with the method using E_2 has the best performance.

In Fig. 2 the number of iterations required for a 16 bit resolution solution is plotted in terms of $\mathrm{Log}(C_1/C_2)$, for two values of C_2. The number of iterations is 5 up to an error of 1000%, except a divergence at $C_1 = C_2$, which make the error matrix itself singular.

From the results shown in Figs. 1 and 2, the different means of generating the error matrix are quite similar in terms of the convergence rate. It is clear that any of the above methods is capable of solving this class of problems. Using an optical processor will limit us to use **E** for the error matrix. When the matrix **A** is written in the analog processor because of the inherent inaccuracies of the processor A^o will be recorded instead of A. But when using this algorithm on a digital computer, we are not limited to any one of the above methods. In this case using E_2 for generating the error matrix will require the least number of operations.



Fg. 2 Number of iterations required for 16 bit resolution in terms of $\mathrm{Log}(C_1/C_2)$ for a 5x5 matrix with rank=4.

4. SEMILINEAR PROBLEMS

This class of problems has a very wide range of applications: such as optimization problems, linear programming, image restoration, etc. In semilinear problems the system of linear equation given in Eq. (1) need to be solved for $\vec{x}$ with the following constraints

$$x_i > 0 \quad , \quad i = 1,\cdots,n \tag{12}$$

and

$$b_i > 0 \ . \tag{13}$$

Usually, the system of linear equations considered in the semilinear problems is overdetermined, underdetermined or with matrix **A** singular. In all these cases the system does not have a unique solution. So in the iterative BOC algorithm any solution that does not satisfy Eq. (12) will be replaced zero, until the system converges to a satisfactory solution.

5. CONCLUSIONS

The BOC algorithm is introduced and applied to solve a set of ill-posed algebra problems. The convergence of the solution was studied for the different methods used to generate the error matrix. The algorithm can be used both optical and digital computers.

6. ACKNOWLEDGEMENTS

This research was supported by the Innovative Science and Technology Office of the Strategic Defense Initiative Organization, administered through The Office of Naval Research

under contract N00014-86-k-0591, and by Rome Air Development Center, administered through Georgia Institute of Technology under contract E-21-669-S23.

REFERENCES

1. J. W. Goodman, A. R. Dias, and L. M. Woody, "Fully Parallel High Speed Incoherent Optical Method for Performing the Discrete Fourier Transform", Opt. Lett. 2, 1 (1978).

2. D. Casasent, "Acousto-Optic Transducers in Iterative Optical Vector-Matrix Processors," Appl. Opt. 21, 1859 (1982).

3. D. Psaltis, D. Casasent, D. Neft, and M. Carlotto, "Accurate Numerical Computation by Optical Convolution," Proc. SIE 232, 160 (1980).

4. M. A. G. Abushagur and H. J. Caulfield, "Speed and Convergence of Bimodal Optical Computers," Opt. Eng. 26(1), 022-027 (1987).

5. M. A. G. Abushagur, H. J. Caulfield, P. M. Gibson and M. Habli, "Superconvergence of Hybrid Optoelectronic Processors," App. Oct. 26, 4906 (1987).

Error Effects on the Processing of Adaptive Array Data Using the BOC

Mustafa A.G. Abushagur
Mohamad Habli

Electrical and Computer Engineering Department
University of Alabama in Huntsville
Huntsville, AL 35899

ABSTRACT

The Bimodal Optical Computer (BOC) is considered for Adaptive Phased Array Radar (APAR) data processing. The effect of the errors in the BOC on the optimum weight calculations for the interference canceling are studied. Computer simulations for five and nine element APARs are presented.

I. INTRODUCTION

The adaptive phased array radar systems provide the means for suppressing unwanted interference signals. This is achieved by nulling the antenna pattern at the direction of the jammers.[1-5] Many algorithms are introduced for the adaptation process and these are reviewed by Monzingo and Miller.[2] These algorithms have large computational complexity. This makes the adaptation process slow, especially for large size arrays.

We have introduced the bimodal optical computer (BOC) for solving linear algebra problems.[6-7] BOC is capable of solving this class of problems with both high speed and accuracy. As an optical hybrid system, the BOC combines between the speed and parallelism of optical analog processors and the accuracy of the digital electronics. In previous paper we have suggested the implementation of the BOC for processing of the APAR's data.[8]

In this paper we study the effect of the errors involved with the BOC components on the interference canceling ability of the APAR. Also a comparison between the antenna pattern for an APAR computed using the BOC with and without optical errors is presented.

1.1 Adaptive Phased Array Radar

The adaptive phased array radar system (APAR) block diagram is shown in Fig.1. The received signal vector $\mathbf{x}$ is given by

$$\mathbf{x} = \mathbf{x}_d + \mathbf{x}_i + \mathbf{x}_n, \tag{1}$$

where $\mathbf{x}_d$ = the desired signal,
$\mathbf{x}_i$ = the interference signal, and
$\mathbf{x}_n$ = the thermal noise.

The received signal x_i's are then multiplied by the corresponding weights w_i's, and summed to give the output signal $s(t)$. The output signal is then compared with the reference signal $r(t)$ to give the error signal $\epsilon(t)$.

$$\epsilon(t) = r(t) - \mathbf{w}^T \mathbf{x}, \tag{2}$$

where ^{T} denotes the transpose of the vector.

The adaptation problem for the weights can be formulated in such a way that the least mean

Presented at the 32nd SPIE Annual Symposium, San Diego, CA August 14-19,1988, SPIE Proceedings 975-30.

square error

$$E\{\epsilon^2(t)\} = E\{[r(t)- \mathbf{w}^T \mathbf{x}]^2\} \tag{3}$$

is minimized, where E{ . } is the ensemble average. In the static case the least mean square error (LMS) can be set to zero. This can be achieved by solving the system of linear equations given by [2],

$$R_{xx} \mathbf{w} = \mathbf{r}_{xd} \ . \tag{4}$$

Where $R_{xx} = E\{\mathbf{x}\,\mathbf{x}^T\}$, and $\mathbf{r}_{xd} = E\{\mathbf{x}_d\, r\}$. So the problem for computing the optimum weights $\mathbf{w}$ is reduced to the solution of a system of linear equations given by Eq. (4). The covariance matrix R_{xx} is symmetrical and usually either singular or ill-conditioned. Solving such system of equations with limited accuracy processor will require a very large number of iterations, which in turn increases the computational time.

1.2 The Bimodal Optical Computer

In this subsection we review briefly the BOC algorithm which is used to solve the adaptation problem given by Eq.(4). Let $A = R_{xx}$, $\mathbf{x} = \mathbf{w}$, and $\mathbf{b} = \mathbf{r}_{xd}$. Substituting in Eq.(4) will reduce it to the following

$$A\mathbf{x} = \mathbf{b} \,. \tag{5}$$

Which the standard format for a system of linear equations. The BOC block diagram is shown in Fig.2. It is an optical-hybrid system. The optical analog processor is used in solving the system of linear equations of Eq.(5) and a digital processor to refine the accuracy of the solution. The algorithm used in solving the system of equations is given as follows:

(a) Solve $A\mathbf{x} = \mathbf{b}$ using the analog optical processor to get $\mathbf{x}_0$.
(b) With a dedicated digital electronics processor, read $\mathbf{x}_0$ and evaluate the residue

$$\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0 = A\,\Delta\mathbf{x}_0 \,. \tag{6}$$

(c) Normalize $\mathbf{r}_0$ to use the dynamic range of the system.
(d) Solve optically the system of equations

$$A\mathbf{z} = s\mathbf{r}_0 \,, \tag{7}$$

where

$$\mathbf{z} = s\,(\Delta\mathbf{x}_0), \tag{8}$$

and

s = radix used in normalizing **r**.
(e) Evaluate electronically

$$\mathbf{x}_1 = \mathbf{x}_0 + \Delta\mathbf{x}_0 \,, \tag{9}$$

and

$$\mathbf{r}_1 = \mathbf{b} - A\mathbf{x}_1 \,. \tag{10}$$

(f) If $\| \mathbf{r}_1 \|$ is small enough stop. Otherwise go back to step (c) and recycle.

There are a number of error sources in the BOC. There are reading and writing errors in **x** and **b** and errors in writing the matrix A. Since the processor is analog, thus inherently it has a low accuracy which is determined by the dynamic range of all of these optoelectronic devices. These errors will affect the rate of convergence of the solution, which also depends on the condition number of the matrix A. In the following section we study the effect of these errors on the performance of the APAR.

II. COMPUTER SIMULATION RESULTS

We have conducted a set of numerical experiments to simulate the BOC in calculating the optimum weights for the APAR. The simulation experiments are considered with and without errors encountered in the BOC. This is done to study the effect of the optical errors in calculating the weights and the antenna pattern. In the following results we have considered the errors to be Gaussian distributed with a standard deviation of 1% in the matrix A and the vectors **x** and **b**.

2.1 Five Element Array

In the first simulation experiment we considered a five element adaptive phased array antenna. The antenna elements are spaced by half a wavelength. In all the cases reported the strength of the desired signal strength is considered to be equal to that of the jammer. In Fig.3 the antenna pattern of the APAR is plotted in terms of the azimuth angle. The desired signal is at zero degrees. In Fig.3(a) the antenna pattern is plotted for a five element array with one jammer at 30°. The pattern was plotted first assuming no errors in the BOC (the continuous curve). The other three curves are for the BOC with the error specified above for 1 iteration (dashed line), 2 iterations (•) and 5 iterations (x). The antenna pattern nulls at 30°, which is the jammer direction. The pattern calculated with only one iteration nulls also at the 30° but not as deep as the exact one, but the 2 and 5 iterations patterns null as deep as the no error curve. In Fig.3(b) the same APAR is considered with a thermal noise of standard deviation equal to 0.1 of the desired signal strength. The continuous curve represents the pattern obtained by the BOC with optical errors, while the dashed line is the BOC with the errors mentioned above. The two patterns are very close except that the dashed one is not as deep as the continuous one, with one or two extra iterations we can expect the same null. Figs.3(c) and (d) the APAR is considered with 2 jammers at 30° and 50°. Again the patterns in Fig.3(c) is without thermal noise, the nulls are very deep and similar with no errors or with a practical BOC with few iterations. In Fig3(d) thermal noise is again considered.

2.2 Nine Element Array

In this set of simulation experiments we consider 9 element APAR with a number of jammers with equal strengths to that of the desired signal which is at 0°. In Fig.4(a) a jammer at 50° is considered and no thermal noise. The pattern is plotted using the BOC with no errors and the BOC with errors for one, and three iterations. From the pattern it is clear that the pattern plotted using three iterations has a null as deep as that of the system with no errors. In Fig.4(b) a thermal noise of standard deviation equal to 0.1 of the desired signal strength is considered with a jammer again at 50°. The antenna pattern obtained using the BOC with no errors and with errors and one iteration are both nulling the jammer with almost the same depth. In Fig.4(c) five jammers are considered at 10°, 20°, 30°, 40°, and 50°, and thermal with strength equal to that of the desired signal. Again the pattern and the depth of the nulls are very similar for BOC with no errors or with errors after few iterations.

III. CONCLUSIONS

The bimodal optical computer because of its hybrid nature is superior in speed to that of the digital computer in solving a system of linear equations, especially for large size systems.[6] It is shown here that the BOC is capable of determining the optimum weights for an APAR in few iterations even when their exists large inaccuracies in the optoelectronic devices used. Since only a very small of number iterations are needed then the weights can be updated very fast which is a must for such system.

IV. ACKNOWLEDGEMENTS

This research is supported by Rome Air Development Center, administered through Georgia Institute of Technology under Contract E-21-669-S23.

REFERENCES

1. B. Widrow, P.E. Manety, L.J. Griffiths AND B.B. Goode,"Adaptive Antenna Systems," Proc. IEEE, **55**, p. 2143 (1967).
2. Robert M. Monzingo and Thomas W. Miller, Introduction to Adaptive Arrays, John Wiley and Sons, New York (1980).
3. Bernard Widrow and Samuel D. Stearns, Adaptive Signal Processing, Prentice-Hall, Englewood Cliffs, New Jersey (1985).
4. S. Haykin, Editor, Array Signal Processing, Prentice-Hall, Englewood Cliffs, New Jersey, (1985).
5. R.T. Compton ,Jr., Adaptive Antennas: Concepts and Performance, Prentice Hall, Englewood Cliffs, New Jersey (1988).
6. M.A.G. Abushagur, and H.J. Caulfield,"Speed and Convergence of Bimodal Optical Computer," Opt. Eng. **26**, p. 022 (1987).
7. M.A.G. Abushagur, H.J. Caulfield. P.M. Gibson, and M. Habli,"Superconvergence of Hybrid Optoelectronic Processors," App. Opt. **26**, p. 4906 (1987).
8. M.A.G. Abushagur,"Adaptive Array Radar Data Processing Using the Bimodal Optical Computer," to be published in Microwave and Optical Technology Letters (1988).

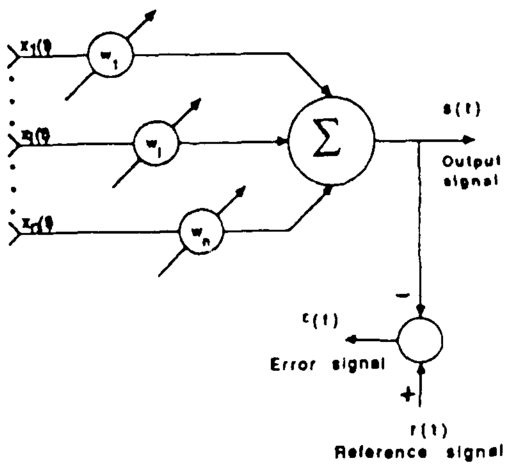


Fig.1 The adaptive phased array radar system.

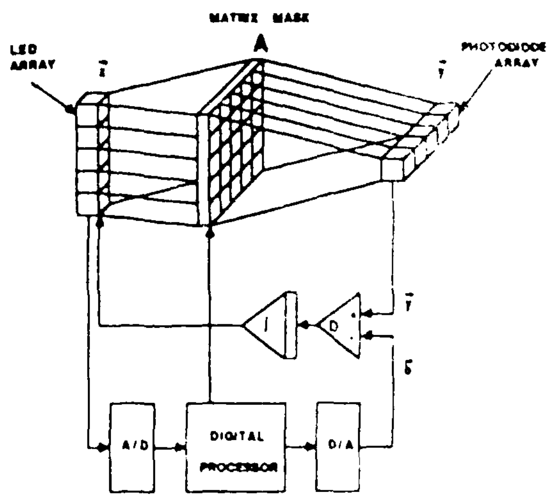


Fig.2 The bimodal optical computer block diagram.

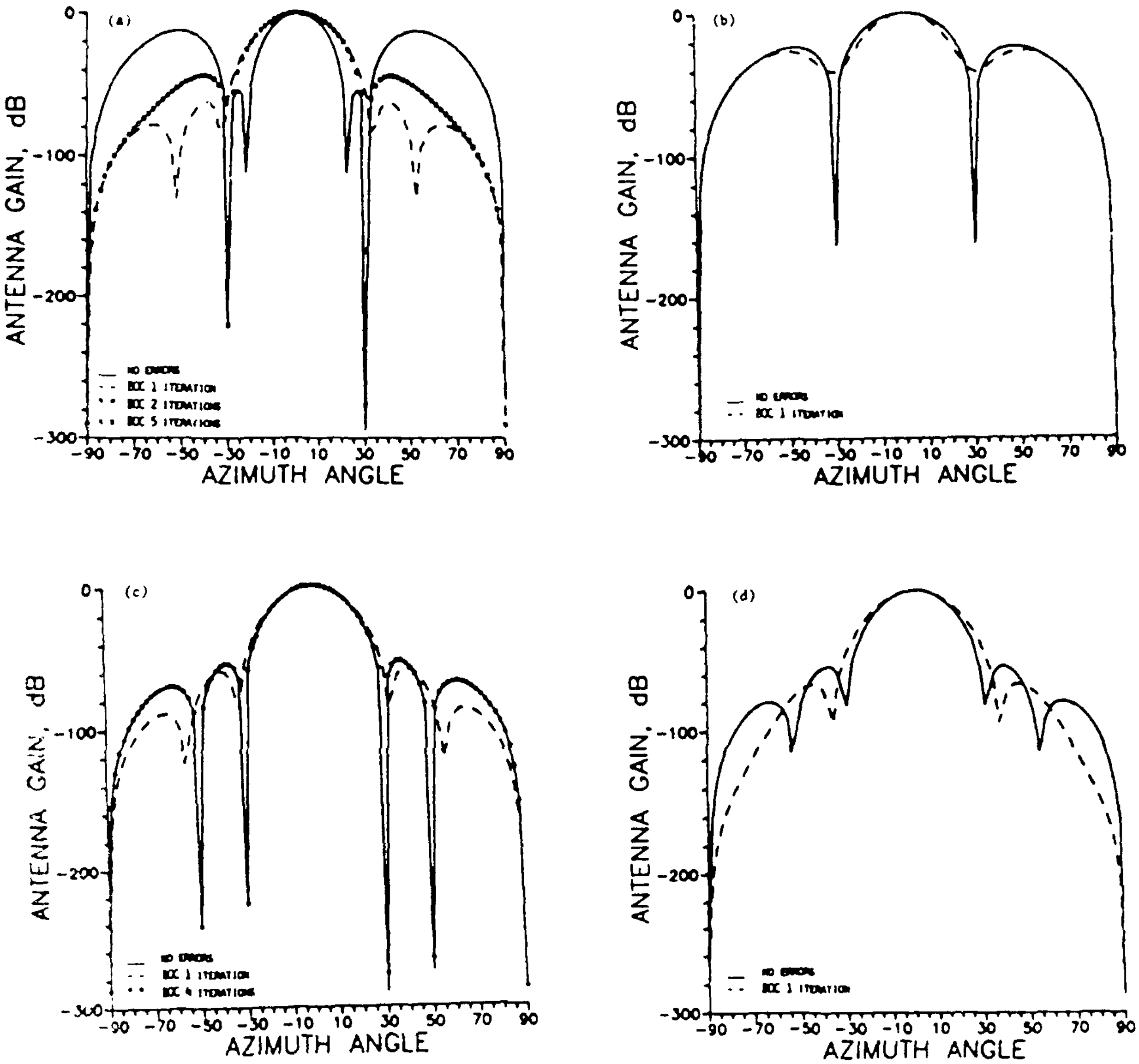


Fig.3 The antenna pattern for a five element array is plotted in terms of the azimuth angle for (a) one jammer at 30° and no thermal noise, (b) one jammer at 30° and thermal noise, (c) two jammers at 30° and 50° and no thermal noise, and (d) as in (c) with thermal noise.

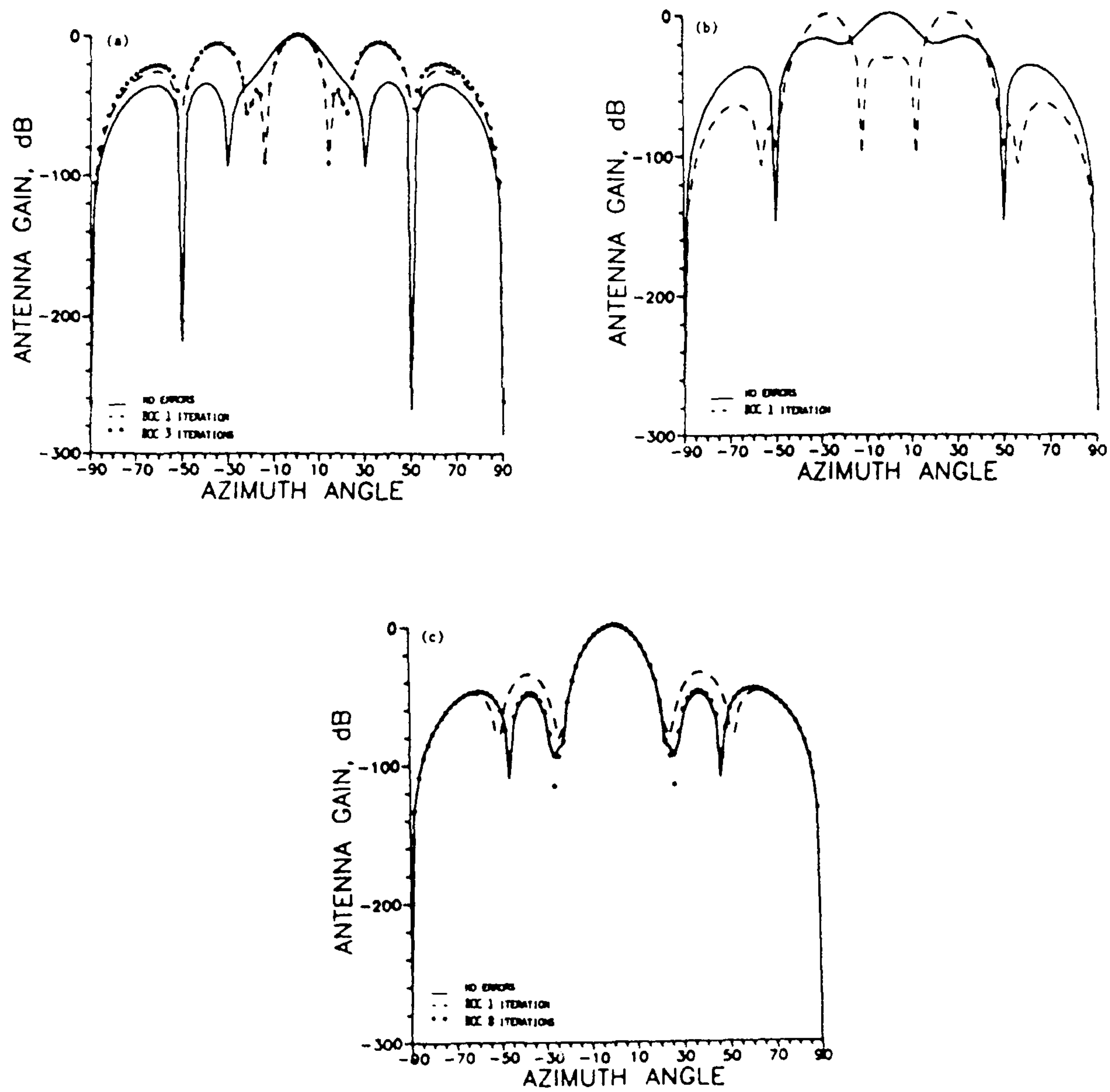


Fig.4 The antenna pattern for a nine element array is plotted for (a) one jammer at 50° with no thermal noise, (b) same as in (c) with thermal noise, and (c) five jammers at 10°, 20°, 30°, 40°, and 50° with thermal noise.

ADAPTIVE ARRAY RADAR DATA PROCESSING USING THE BIMODAL OPTICAL COMPUTER

Mustafa A. G. Abushagur
Electrical and Computer Engineering Department
University of Alabama in Huntsville
Huntsville, Alabama 35899

KEY TERMS

Adaptive arrays, optical computing, optical data processing

ABSTRACT

The use of the bimodal optical computer (BOC) in determining the weights for an adaptive phased array radar is introduced. Interference canceling is presented for two cases: (1) assuming the direction of the jammer is known and (2) assuming no a priori information. The effect of the jammers on the array pattern is shown for up to four jammers.

1. INTRODUCTION

The sensitivity of a signal-receiving antenna array system to interfering noise sources can be reduced by suitable processing of the outputs of the individual array elements. The processing of the output of the array system acts as an adaptive filtering system [1–4]. The adaptive phase array radar systems provide the means of suppressing unwanted interference signals. This is achieved by nulling the array pattern in the direction of the jammers. Many algorithms have been introduced for the adaptation process and they are reviewed by Monzingo and Miller [2].

In this paper we present a new technique for determining the weights for the adaptive array using the bimodal optical computer (BOC) [5–7]. The bimodal optical computer is capable of solving systems of linear equation very rapidly with high accuracy. In the adaptation process we reduce the problem to a system of linear equations, which in turn is solved using the BOC.

In Section 2 we review the basic theory of adaptive phased array radars. The bimodal optical computer algorithm for solving the adaptation problem is presented in Section 3. Computer simulation results are given in Section 4. Conclusions and final remarks are given in Section 5.

2. ADAPTIVE PHASED ARRAYS

In adaptive phased array radars the incoming signal is detected by an array of sensors. The detected signal is a combination of the target signal plus interference and noise signals. The system is adjusted in such a way to suppress the interference signal reception without affecting the desired signal.

In this section we consider the two general cases of interference canceling: (1) by assuming that the interference signal direction is known and (2) by assuming no a priori information is known about the interference signal.

2.1. Interference Signal Direction is Known. When the interference signal direction is known the weights w of the array can be chosen to suppress the interference signal. Let the system shown in Figure 1(a) be used to demonstrate this adaptation technique. The output signal of the array $s(t)$ is given by [1]

$$\begin{aligned} s(t) = {} & P\left[(w_1 + w_3)\sin\omega_0 t + (w_2 + w_4)\sin(\omega_0 t - \theta - \tfrac{1}{2}\pi)\right] \\ & + I\left[w_1 \sin(\omega_0 t - \theta) + w_2 \sin(\omega_0 t - \theta - \tfrac{1}{2}\pi) \right. \\ & \left. + w_3 \sin(\omega_0 t + \theta) + w_4 \sin(\omega_0 t + \theta - \tfrac{1}{2}\pi)\right], \end{aligned} \quad (1)$$

where

$$\begin{aligned} P &= \text{the pilot signal,} \\ I &= \text{the interference signal,} \\ \theta &= \text{the phase shift,} \\ \theta &= \frac{2\pi d}{\lambda}\sin\phi \end{aligned} \quad (2)$$

To cancel the interference signal and to make the signal $s(t)$ equal to the pilot signal, we need to solve the following system of linear equations for the weights w

$$\begin{aligned} w_1 + w_3 &= 1, \\ w_2 + w_4 &= 0, \\ (w_1 + w_3)\cos\theta - (w_2 - w_4)\sin\theta &= 0, \\ (w_2 + w_4)\cos\theta + (w_1 - w_3)\sin\theta &= 0 \end{aligned} \quad (3)$$

The size of this system of linear equations depends on the number of sensors in the array. The number of jammers can make the system under or overdetermined, both of which are time consuming algebra problems.

2.2. No A Priori Information is Known. This is the most general case where we assume no information about jammers. The system used in this case is shown in Figure 1(b). Each of the n sensors receives a signal $x_i(t)$ that is in turn multiplied by a variable weight w_i. The output signal $s(t)$ is compared with the desired signal $d(t)$ and their difference, the error signal $\epsilon(t)$, is used to determine the value of w_i. The output of the array is

$$s(t) = \sum_{i=1}^{n} x_i(t) w_i \quad (4)$$

or

$$s(t) = \mathbf{w}^T \mathbf{x}, \quad (5)$$

where

$$\mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} \quad \text{and} \quad \mathbf{x} = \begin{bmatrix} x_1(t) \\ x(t) \\ x_n(t) \end{bmatrix} \quad (6)$$

For digital sampled data

$$s(j) = \mathbf{w}^T \mathbf{x}(j) \quad (7)$$

and

$$\epsilon(j) = d(j) - s(j) = d(j) - \mathbf{w}^T \mathbf{x}(j) \quad (8)$$

The optimum value of the weights w is the one that reduces $\epsilon(j)$ to zero or at least minimizes it.

For N samples of data the optimum weights satisfy the following set of systems of linear equations

$$\begin{aligned} \mathbf{w}^T \mathbf{x}(1) &= d(1) \\ & \vdots \\ \mathbf{w}^T \mathbf{x}(i) &= d(i) \\ \mathbf{w}^T \mathbf{x}(N) &= d(N) \end{aligned} \quad (9)$$

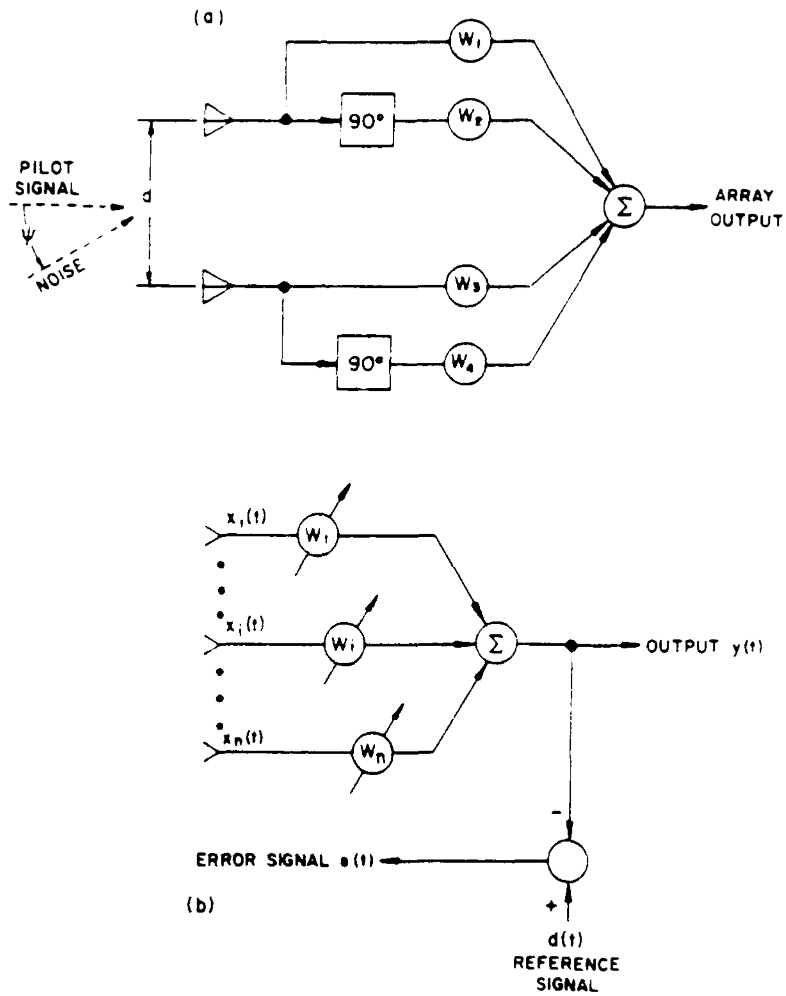


Figure 1 Basic adaptive array system with (a) signal and noise directions known and (b) no a priori information assumed

The N sets of equations have n unknowns, and usually $N \gg n$, and are inconsistent and overspecified. The optimization problem can be rewritten as

$$w_{opt} = R_{xx}^{-1} r_{xd} \tag{10}$$

where

$$R_{xx} = E\{x x^T\} \tag{11}$$

and

$$r_{xd} = E\{x d\} \tag{12}$$

The matrix R_{xx} is called the covariance matrix, where $E\{\ \}$ is the ensemble average.

Many algorithms are introduced [2] to solve for the weights in Eq. (10). Some of the popular algorithms are the least mean square (LMS) and the direct matrix inversion (DMI).

We will briefly mention the DMI algorithm since it leads to the algorithm introduced in this paper. Equation (10) cannot be determined exactly using a limited number of samples of the input data. For practical consideration a small number of samples is detected to be used in determining w. The estimated value of Eq. (10) can be given by

$$\hat{w} = \hat{R}_{xx}^{-1} \hat{r}_{xd} \tag{13}$$

where $\hat{R}_{xx}$ is the sample covariance matrix and $\hat{r}_{xd}$ is the sample cross-correlation vector that are given by

$$\hat{R}_{xx} = \frac{1}{K} \sum_{j=1}^{K} x(j) x^T(j) \tag{14}$$

and

$$\hat{r}_{xd} = \frac{1}{K} \sum_{j=1}^{K} x(j) d(j) \tag{15}$$

K is the number of samples. The DMI algorithm determines the inverse of the sample covariance matrix $\hat{R}_{xx}$ and then from Eq. (13) evaluates $\hat{w}$.

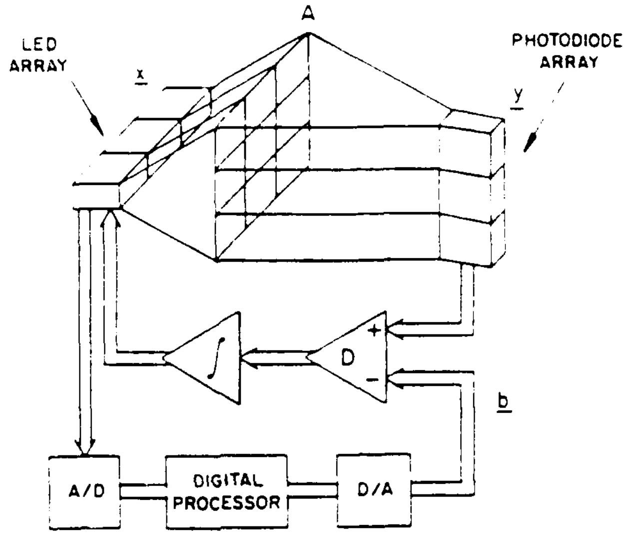


Figure 2 The bimodal optical computer used in solving a system of linear equations

3. THE BIMODAL OPTICAL COMPUTER ALGORITHM

Convergence of either the LMS or the DMI algorithms depends on a number of factors, the most important being the condition number of the matrix $\hat{R}_{xx}$. If the matrix $\hat{R}_{xx}$ is ill-conditioned or singular, it either converges very slowly or the inverse does not exist, respectively. In such cases other methods might be used, but they are lengthy and time consuming, so they are not suitable for a system where time is a very crucial element.

We have shown in previous publications [6, 7] that the bimodal optical computer is capable of solving such problems, where the system of equations is ill-conditioned, singular, overspecified, or underspecified. The BOC is a hybrid system by nature; see Figure 2. It uses analog optics to solve the problem approximately but rapidly and it utilizes the digital electronics to refine the solution, in an iterative scheme.

The adaptation problem for the weights w introduced in Section 2, can be rewritten in the following form, from Eq. (13),

$$\hat{R}_{xx}\hat{\mathbf{w}} = \hat{\mathbf{r}}_{xd}, \qquad (16)$$

which can be written as

$$A\mathbf{x} = \mathbf{b}, \qquad (17)$$

where

$$A = \hat{R}_{xx}$$
$$\mathbf{x} = \hat{\mathbf{w}}$$
$$\mathbf{b} = \hat{\mathbf{r}}_{xd}$$

Equation (16) is a system of linear equations that can be solved using the bimodal optical computer. Among the advantages of using the BOC over the conventional techniques are speed (especially for large size arrays), convergence of the solution for difficult problems, and ill-conditioned singular systems which is the case for most of the adaptive array radar problems.

We review here the BOC algorithm in solving the system $A\mathbf{x} = \mathbf{b}$

(a) Solve $A\mathbf{x} = \mathbf{b}$ using the analog optical processor to get $\mathbf{x}_0$.

(b) With a dedicated digital electronics processor, read $\mathbf{x}_0$ and evaluate the residue

$$\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0 = A\mathbf{x} - A\mathbf{x}_0 = A(\Delta\mathbf{x}_0) \qquad (18)$$

(c) Normalize $\mathbf{r}_0$ to use the dynamic range of the system

(d) Solve optically the system

$$A\mathbf{z} = s\mathbf{r}_0 \qquad (19)$$

where

$$\mathbf{z} = s(\Delta\mathbf{x}_0), \qquad (20)$$

and s is the radix used in normalizing $\mathbf{r}$

(e) Evaluate electronically

$$\mathbf{x}_1 = \mathbf{x}_0 + \Delta\mathbf{x}_0 \qquad (21)$$

and

$$\mathbf{r}_1 = \mathbf{b} - A\mathbf{x}_1 \qquad (22)$$

(f) If $\mathbf{r}_1$ is small enough, stop. Otherwise, go to (c) and recycle

In the following section we present some of the preliminary results from computer simulation studies of the BOC in processing adaptive array problems

4. SIMULATION RESULTS

Two simulation experiments are presented in this section. In the first experiment we used a five element array and assume

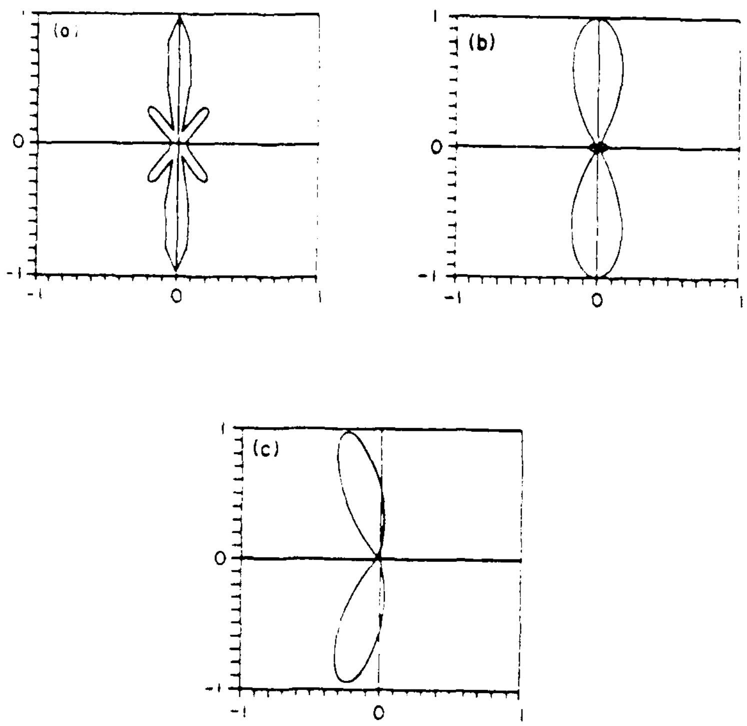


Figure 3 Phased array pattern for five elements (a) before adaptation, (b) adapted for a jammer at 45°, and (c) adapted for four jammers at 45°, 80°, 120°, and 150

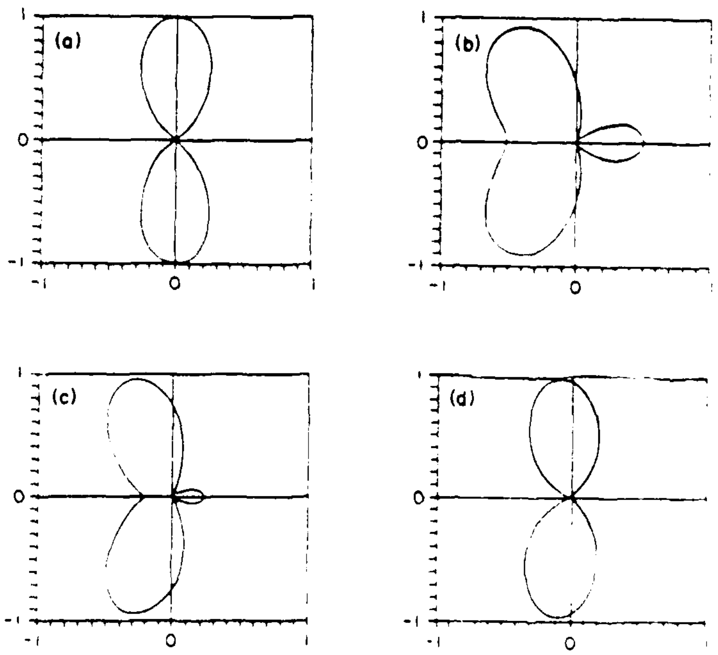


Figure 4 Two element phased array pattern (a) before adaptation and (b)–(d) adapted for single jammers at [illegible] 45 and [illegible] respectively

the directions of the jammers were known. In the second experiment a two element array is used and no a priori information is assumed.

In Figure 3 the five element array pattern is plotted as a function of the angle ψ. Figure 3(a) shows the array pattern before adaptation. In Figure 3(b) the pattern after adaptation is shown for a jammer at 45°. The array pattern after adaptation has reformed in such a way that it nulls the jammer signal. In Figure 3(c) four jammers are considered at 45°, 80°, 120°, and 150°. The array pattern is again reformed to null all the jammers signal reception.

In Figure 4 the BOC was used to solve the adaptation problem assuming no a priori information about the interference signals. Figure 4(a) shows the two-element array pattern before adaptation. In Figure 4(b)–(d) the pattern is plotted for a single jammer placed at 30°, 45°, and 60°, respectively. In all these plots the array adapted to cancel the interference signal in each of the given cases. In all of the preceding results the jammer signals are considered to be of the same strength as the desired signal, and the convergence of the solution obtained in less than five iterations. Also the condition numbered of the R_{ii} is between 10^6 and ∞.

5. CONCLUSIONS

The bimodal optical computer is shown in these preliminary results to present a powerful mean for solving adaptive phased array problems. We are considering in future work larger array sizes, receiver noise, and very strong interference signals.

ACKNOWLEDGMENTS

The author would like to thank Dr. H. John Caulfield for numerous enlightening discussions.

This research was supported by the Innovative Science and Technology Office of the Strategic Defense Initiative Organization, administered through the Office of Naval Research under contract N00014-86-k-0591, and by Rome Air Development Center, administered through Georgia Institute of Technology under contract E-21-669-S23.

REFERENCES

1. B. Widrow, P. E. Mantey, L. J. Griffiths, and B. B. Goode, "Adaptive Antenna Systems," *Proc. IEEE*, vol. 55, 1967, p. 2143.
2. R. M. Monzingo and T. W. Miller, *Introduction to Adaptive Arrays*, Wiley, New York, 1980.
3. B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.
4. S. Haykin ed., *Array Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.
5. H. J. Caulfield, J. H. Gruninger, J. E. Ludman, K. Steiglitz, H. Rabitz, J. Gelfand, and E. Tsoni, "Bimodal Optical Computers," *Appl. Opt.*, Vol. 25, 1986, p. 3128.
6. M. A. G. Abushagur and H. J. Caulfield, "Speed and Convergence of Bimodal Optical Computers," *Opt. Eng.*, Vol. 26, 1987, p. 22.
7. M. A. G. Abushagur, H. J. Caulfield, P. M. Gibson, and M. Habli, "Superconvergence of Hybrid Optoelectronic Processors," *Appl. Opt.*, Vol. 26, 1987, p. 4906.

Received 6-20-88

Microwave and Optical Technology Letters, 1/7, 236–240

CCC 0895-2477/88/$4.00

MISSION

of

Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control, Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of competence is provided to ESD Program Offices (POs) and other ESD elements to perform effective acquisition of C^3I systems. The areas of technical competence include communications, command and control, battle management information processing, surveillance sensors, intelligence data collection and handling, solid state sciences, electromagnetics, and propagation, and electronic reliability/maintainability and compatibility.